Privera.

Privera säkerhetsunderlag

Dokumentation om säkerhet, sårbarhetshantering, loggning, underleverantörer och livscykel, för IT- och informationssäkerhetsfunktioner.

Version	Privera 0.9.0-beta23
Datum	2026-11-12
Tillverkare	Itha AB, org.nr 559282-3826
Adress	Bränningestrandsvägen 40b, 151 39 Södertälje, Sverige
Webb	privera.se/security/it/
Säkerhet	feedback@privera.se
Dataskyddsfrågor	info@privera.se
Support	support@privera.se, 076 926 35 70

Dokumenten finns också som webbsidor på privera.se/security/it/, där den senaste versionen alltid ligger. Avtalsbilagan om säkerhet, incidenter, revision och insyn är ett utkast för juristgranskning och lämnas ut på begäran till info@privera.se.

Innehåll

1. IT-drift 3
Tyst installation per dator, Intune, AppLocker och WDAC, registerpolicyer med ADMX-mall, styrning av uppdateringar, VDI och avinstallation.

2. Sårbarhetshantering 13
Mottagning, bedömning med CVSS, tidsramar för åtgärd, samordnat röjande och säkerhetsmeddelanden.

3. Incidentrutin 17
Roller, vad som räknas som incident, förlopp och rapportering till CERT-SE, ENISA, IMY och kunder.

4. Säkerhetsloggning 21
Händelsekoder, hashkedja, Windows händelselogg, registerpolicy och insamling till SIEM.

5. Kryptering och åtkomstkontroll 34
TLS, kryptering i vila, nyckelhantering och varför programmet saknar egna konton och roller.

6. Underleverantörer 38
Vilka leverantörer som behandlar vilka uppgifter, var de finns och på vilken överföringsgrund.

7. Livscykel och support 42
Supportperiod, säkerhetsuppdateringar, uppdateringskanal, drift utan licensserver och slut på support.

8. Säker utveckling 46
Tester, releasekontroller, signering, beroenden och det som ännu saknas.

9. SBOM 52
Komponentförteckningen i CycloneDX: vad den täcker, hur den tas fram och kända luckor.

10. Avvecklingsläge 55
Hur Privera fortsätter fungera utan licensservern om Itha AB upphör eller slutar med Privera.

11. Kontroller enligt ISO 27001 59
Hur rutinerna förhåller sig till kontrollerna i ISO/IEC 27001:2022, varför certifieringen väger lättare för ett lokalt program, och underleverantörernas intyg.

12. Plan för CRA 63
Läget mot EU:s cyberresiliensförordning: klassning, rapportering, supportperiod och tidsplan.

13. Stöd för kundens NIS2-efterlevnad 67
Hur Privera och Itha AB stöder kundens arbete enligt NIS2 och cybersäkerhetslagen, artikel 20, 21 och 23, och vad som saknas.

Komponentförteckningen (SBOM) för version 0.9.0-beta23 finns som maskinläsbar fil på privera.se/security/sbom/privera-0.9.0-beta23.cdx.json, och beskrivningen av den på privera.se/security/it/sbom/. Policyn för sårbarhetsrapporter finns på privera.se/security/sarbarheter/.

IT-drift

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/it-drift/

Beskriver hur en IT-avdelning installerar Privera centralt, styr programmet med grupprincip eller Intune och avinstallerar det. Installern och dess växlar gäller version 0.9.0-beta23. Styrningen av uppdateringar (avsnitt 5) och värdet `UpdateMode` i ADMX-mallen (avsnitt 6) gäller från 0.9.0-beta21. AppLocker-regeln som nekar äldre versioner (avsnitt 4), de signerade tilläggen i installern (avsnitt 2.1), den fullständiga signaturkontrollen av uppdateringar (avsnitt 5) och värdena `RequireEventLogForExport` och `AllowDebugPort` (avsnitt 6) gäller från 0.9.0-beta22. Det som inte är provat på en riktig dator eller i ett riktigt utrullningsverktyg står utskrivet, och är samlat i avsnitt 9.

1. Sammanfattning

Del	Läge
Tyst installation	Stöds: `/S`, per användare (förval) eller per dator (`/allusers`). Växlarna kommer från installerns mall och är inte provade i Intune eller Configuration Manager.
Installation per dator	Stöds, som administratör eller SYSTEM. Programmet hamnar i `C:\Program Files\Privera` och körs sedan utan administratörsrätt.
MSI-paket	Finns inte. Installern är en signerad exe-fil som Intune och Configuration Manager kan köra tyst.
Styrning av uppdateringar	Från 0.9.0-beta21. Policyn `UpdateMode` stänger av uppdateringarna eller låter programmet bara meddela att en ny version finns (avsnitt 5).
Registerpolicyer och ADMX-mall	Sju värden under `HKLM\Software\Policies\Privera`, och en ADMX-mall på svenska och engelska som täcker alla sju (avsnitt 6).
AppLocker och WDAC	Alla programfiler är signerade. Från 0.9.0-beta22 har `Privera.exe` och installern en binär filversion som stiger med varje version (0.9.0.22 i 0.9.0-beta22, 0.9.0.23 i 0.9.0-beta23), och Electrons spärrar i programfilen är satta. En tillåtande utgivarregel för Itha AB släpper också in 0.9.0-beta21 och äldre, där den signerade programfilen kan användas för att köra annan kod. AppLocker behöver därför också en nekande regel för Privera till och med version 0.9.0.21 (avsnitt 4). Inte provat under en tillämpad policy.
VDI och terminalserver	Fungerar tekniskt, men licensen binds till Windows-kontot och datornamnet. Varje konto behöver en egen licens, och poolade skrivbord med växlande datornamn fungerar inte i praktiken (avsnitt 8).
Nätverk	Utgående HTTPS till licens.privera.se och, första gången språkmodellen behövs, till Hugging Face. Ingen inkommande port. Se Underlag för IT och informationssäkerhet.

2. Installation

2.1 Installationsfilen

Installern heter `Privera-Setup-<version>.exe` och hämtas från privera.se/download/ med en licensnyckel. Samma fil installerar per användare och per dator. Den är signerad av Itha AB. Kontrollera signaturen före utrullning:

```
Get-AuthenticodeSignature .\Privera-Setup-0.9.0-beta23.exe | Format-List Status, SignerCertificate
```

`Status` ska vara `Valid` och certifikatets subjekt `CN=Itha AB, O=Itha AB, L=Södertälje, S=Stockholm, C=SE`. Certifikatet kommer från Azure Trusted Signing och byts dagligen. Signaturen har en tidsstämpel och gäller även efter att certifikatet har gått ut. Installern för 0.9.0-beta23 har den binära filversionen 0.9.0.23, liksom `Privera.exe` (avsnitt 4.1).

Smart App Control. Installern packar upp sina tillägg (bland annat `System.dll` och `nsis7z.dll`) i `%TEMP%` och laddar dem där. Från 0.9.0-beta22 är tilläggen i installern och avinstalleraren signerade av Itha AB, och bygget stoppas om något av dem inte är giltigt signerat. I 0.9.0-beta21 och äldre var de osignerade. På en dator med Smart App Control påslaget stoppades då tillägget, och installern avslutades med kod 0 utan att lägga in några programfiler: ingen

`Privera.exe`, men en genväg på skrivbordet. Det är bekräftat i Windows Sandbox 2026-10-23, som kör med Smart App Control påslaget. Där installerades 0.9.0-beta22 med de signerade tilläggen utan fel, och analys, OCR och export fungerade. Kontrollera efter en utrullning att `Privera.exe` finns, och inte bara att installern gav kod 0.

2.2 Per användare eller per dator

	Per användare (förval)	Per dator
Tyst kommando	`Privera-Setup-0.9.0-beta23.exe /S`	`Privera-Setup-0.9.0-beta23.exe /S /allusers`
Kräver	Inget administratörskonto	Administratör eller SYSTEM. Från ett vanligt konto begär installern förhöjning (UAC).
Programmapp	`%LOCALAPPDATA%\Programs\Privera`	`C:\Program Files\Privera`
Registernycklar	Under `HKEY_CURRENT_USER`	Under `HKEY_LOCAL_MACHINE`, 64-bitarsvyn
Genvägar	Användarens Start-meny och skrivbord	Start-menyn och skrivbordet för alla användare
AppLockers standardregler	Stoppar programmet	Tillåter programmet
Automatisk uppdatering	Fungerar utan administratörsrätt	Begär administratörsrätt (avsnitt 5)

Rekommendation för en organisation: installera per dator, sätt `UpdateMode` till 0 och rulla ut nya versioner med samma verktyg som installationen.

Växlarna:

- `/S` kör installern utan fönster. Skriv stort S.
- `/allusers` installerar per dator och `/currentuser` per användare. Utan någon av dem installeras programmet per användare, utom när det redan finns en installation per dator. Då uppgraderas den.
- `/D=C:\Annan mapp\Privera` väljer en annan programmapp. Växeln ska stå sist och skrivas utan citattecken, även om sökvägen har mellanslag.

Det här gäller också, enligt installerns mall:

- Installern avslutar ett Privera som är igång, utan att fråga. En granskning som inte är sparad går då förlorad. Vid installation per dator gäller det alla användares Privera från samma programmapp, även på en terminalserver. Rulla ut när programmet inte används.
- En ny version installeras ovanpå den gamla: installern tar först bort den tidigare versionen och lägger sedan in den nya. Användarnas data i `%APPDATA%` rörs inte.
- Installern begär ingen omstart och startar inte programmet efter en tyst installation.
- Installern registrerar inte händelsekällan `Privera` i Windows händelselogg. Gör det i utrullningen om programmet körs under AppLocker eller WDAC, se avsnitt 2.5 och Säkerhetsloggning avsnitt 7.3.

2.3 Intune (Win32-app)

Packa installern med Microsofts verktyg Win32 Content Prep Tool:

```
IntuneWinAppUtil.exe -c C:\Paket\Privera -s Privera-Setup-0.9.0-beta23.exe -o C:\Paket\ut
```

Fält	Värde
Installationskommando	`Privera-Setup-0.9.0-beta23.exe /S /allusers`
Avinstallationskommando	`"C:\Program Files\Privera\Uninstall Privera.exe" /allusers /S`
Installationsbeteende	System

Fält	Värde
Omstartsbeteende	Ingen särskild åtgärd
Returkod	0 = lyckades
Identifieringsregel	Register, se nedan

Identifieringsregel:

Fält	Värde
Regeltyp	Register
Nyckelsökväg	`HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\dfe606d1-6622-5834-8145-638a6cb1ca05`
Värdenamn	`DisplayVersion`
Identifieringsmetod	Strängjämförelse
Operator	Lika med
Värde	`0.9.0-beta23`
Associerad med en 32-bitarsapp på 64-bitarsklienter	Nej

Nyckelns namn är programmets fasta id och är detsamma i alla versioner. Installern skriver där också `DisplayName` (Privera och versionen), `UninstallString` och `QuietUninstallString`. Nyckeln och värdena är avlästa på en dator med programmet installerat per användare, där samma nyckel ligger under `HKEY_CURRENT_USER`. Den binära filversionen i `Privera.exe` är 0.9.0.23 i 0.9.0-beta23 och 0.9.0.22 i 0.9.0-beta22, men 0.9.0.0 i alla tidigare betaversioner, så den skiljer inte äldre versioner åt. Identifieringsregeln bygger därför på `DisplayVersion`.

En ny version läggs ut som en ny Win32-app som ersätter den gamla, eller genom att byta fil och identifieringsregel i den befintliga appen. Den tidigare versionen behöver inte avinstalleras först, eftersom installern uppgraderar på plats.

Per användare: installationsbeteende Användare, kommandot `Privera-Setup-0.9.0-beta23.exe /S /currentuser` och samma nyckel under `HKEY_CURRENT_USER`. Programmet hamnar då i användarens profil, med följderna i avsnitt 4.

Inte provat i Intune.

2.4 Configuration Manager

Skapa ett program med distributionstypen Skriptinstallation, samma installations- och avinstallationskommando som för Intune, identifieringsmetoden register (`DisplayVersion` lika med versionen, under samma nyckel) och installationsbeteendet Installera för system. Inte provat i Configuration Manager.

2.5 Exempel på installationsskript

Ett skript som installerar per dator och registrerar händelsekällan. Körs som SYSTEM, till exempel med installationskommandot `powershell.exe -ExecutionPolicy Bypass -File .\installera.ps1`:

```
$p = Start-Process -FilePath "$PSScriptRoot\Privera-Setup-0.9.0-beta23.exe" -ArgumentList '/S', '/allusers' -
Wait -PassThru
if ($p.ExitCode -ne 0) { exit $p.ExitCode }
New-EventLog -LogName Application -Source Privera -ErrorAction SilentlyContinue
exit 0
```

Inte provat.

3. Avinstallation

Installation	Tyst avinstallation
Per dator	`"C:\Program Files\Privera\Uninstall Privera.exe" /allusers /S`
Per användare	`"%LOCALAPPDATA%\Programs\Privera\Uninstall Privera.exe" /currentuser /S`

Kommandona är de som installern själv skriver i `QuietUninstallString`. Avinstallationen avslutar Privera om det är igång.

Koppla loss licensen först, under Inställningar, Licens, så att den kan användas på en annan dator direkt. Annars får den nya datorn ta över licensen, och en övertagning går högst en gång var 30:e dag.

Kvar efter avinstallationen, för varje Windows-konto som har använt programmet:

Plats	Innehåll
`%APPDATA%\privera-desktop\`	Licensfilen och sparade granskningar, krypterade med Windows DPAPI. Granskningarna innehåller uppgifter ur de dokument som analyserats. Exportloggar i `audit\`, okrypterade och med filnamn och sökvägar. Säkerhetsloggen i `sakerhetslogg\` och dess ankare. Inställningar och webbläsarcache.
`%LOCALAPPDATA%\privera-desktop-updater\`	Hämtade uppdateringar.
`%USERPROFILE%\.cache\huggingface\hub\models--KB--bert-base-swedish-cased-ner\`	Språkmodellen, cirka 500 MB.

Kvar blir också säkerhetsloggen i den katalog som policyn anger, posterna i Windows händelselogg samt exporterade filer och de exportloggar som sparats bredvid dem. Ta inte bort säkerhetsloggen om den behövs för uppföljning.

Så rensas ett konto, som den användaren:

```
Remove-Item "$env:APPDATA\privera-desktop", "$env:LOCALAPPDATA\privera-desktop-updater",
"$env:USERPROFILE\.cache\huggingface\hub\models--KB--bert-base-swedish-cased-ner" -Recurse -Force -ErrorAction
SilentlyContinue
```

4. AppLocker och WDAC

4.1 Signaturerna

Programfiler som Itha AB levererar utan egen signatur signeras av Itha AB med tidsstämpel. Filer som redan har en giltig signatur från sin tillverkare behåller den. I version 0.9.0-beta20 fanns 200 programfiler (.exe, .dll, .pyd, .node): 162 signerade av Itha AB, 25 av Python Software Foundation, 11 av Microsoft och 2 av Intel. Från 0.9.0-beta22 står varje programfil i SBOM:en med SHA-256 och uppgift om den bär en Authenticode-signatur, se SBOM.

AppLocker läser utgivaren som `O=ITHA AB, L=SÖDERTÄLJE, S=STOCKHOLM, C=SE`. En AppLocker-utgivarregel jämför utgivarens namn, inte certifikatet, och påverkas därför inte av att certifikatet byts dagligen. Så här visar AppLocker programmets exe-filer (`Get-AppLockerFileInformation`, avläst 2026-10-23):

Fil	Produktnamn	Filnamn	Binär version
`Privera.exe`	`PRIVERA`	tomt (originalfilnamn saknas)	0.9.0.23 i 0.9.0-beta23, 0.9.0.22 i 0.9.0-beta22, 0.9.0.0 i tidigare versioner
`Privera-Setup-<version>.exe` (installern)	`PRIVERA`	tomt	Samma som `Privera.exe`
`Uninstall Privera.exe` (avinstalleraren)	`PRIVERA`	tomt	Samma som `Privera.exe`
`resources\sidecar\sidecar.exe` (motorn)	tomt	tomt	0.0.0.0 (ingen versionsresurs)

Fil	Produktnamn	Filnamn	Binär version
`resources\tesseract\bin\tesseract.exe` (OCR-motorn)	tomt	tomt	0.0.0.0 (ingen versionsresurs)
`resources\elevate.exe`	`ELEVATE APPLICATION`	`ELEVATE.EXE`	1.0.0.2894
`resources\sidecar\_internal\torch\bin\protoc.exe` (del av PyTorch)	`PROTOCOL BUFFERS - GOOGLE'S DATA INTERCHANGE FORMAT`	tomt	3.21.12.0

Äldre versioner kan köra annan kod. Versionerna 0.9.0-beta12 till 0.9.0-beta21 är giltigt signerade av Itha AB, men i deras `Privera.exe` är Electrons inbyggda spärrar (fuses) inte satta. Med miljövariabeln `ELECTRON_RUN_AS_NODE` kör den signerade programfilen då godtycklig kod, och `NODE_OPTIONS` och `--inspect` ger andra vägar till samma sak. En tillåtande utgivarregel för Itha AB, till exempel den som stod i det här dokumentet före 0.9.0-beta22, släpper in de versionerna och därmed den vägen till egen kod. Lägg därför till den nekande regeln i avsnitt 4.2.

Spärrarna från 0.9.0-beta22. I `Privera.exe` är körning som Node (`RunAsNode`), `NODE_OPTIONS` och `--inspect` avstängda. Asar-integriteten och `OnlyLoadAppFromAsar` är påslagna, så programmet startar bara med det `app.asar` som bygget skrev, och kakor i programmets profil krypteras. Bygget stoppas om någon spärr har ett annat läge än det förväntade. Spärren som ger `file:` utökade rättigheter (`GrantFileProtocolExtraPrivileges`) står kvar på, eftersom programfönstret laddas från en fil och inte laddas alls utan den.

Felsökningsporten. Startas kundutgåvan med `--remote-debugging-port`, `--remote-debugging-pipe`, `--inspect`, `--inspect-brk` eller `--inspect-port` avslutas den innan något fönster öppnas. Undantaget är när policyn `AllowDebugPort` är 1 (avsnitt 6). Då loggas händelsen `app.felsokningsport_tillaten` (1006).

Kvar: installation per användare. Asar-integriteten skyddar bara `app.asar`. I en installation per användare, i `%LOCALAPPDATA%\Programs\Privera`, äger användaren programmappen och kan byta ut DLL-filer, filerna i `resources\app.asar.unpacked` (bland annat `.node`-filer) och motorn `sidecar.exe`. Det som stoppar det är AppLocker eller WDAC som också kontrollerar DLL-filer: en utbytt fil som är osignerad eller signerad av någon annan stoppas då. En äldre fil med en giltig signatur från samma utgivare stoppas inte av en utgivarregel. En installation per dator, i `C:\Program Files\Privera`, är skrivskyddad för vanliga användare.

4.2 AppLocker

AppLockers standardregler låter vanliga användare köra program bara under `%WINDIR%` och `%PROGRAMFILES%`. En installation per dator i `C:\Program Files\Privera` omfattas av dem. En installation per användare hamnar i `%LOCALAPPDATA%\Programs\Privera` och stoppas, liksom installern om användaren själv kör den. Använd installation per dator, eller lägg till reglerna nedan.

Två regler. Den första tillåter alla programfiler som Itha AB har signerat, oavsett produkt, filnamn och version. Den behövs vid installation per användare och för att motorn och OCR-motorn ska få köras: `sidecar.exe` och `tesseract.exe` saknar versionsresurs, också i 0.9.0-beta23, så AppLocker ser varken produktnamn eller version (`O=ITHA AB, L=SÖDERTÄLJE, S=STOCKHOLM, C=SE\\,0.0.0.0`), och en regel som kräver produktnamn eller lägsta version stoppar dem. Då fungerar inte Privera.

Den andra regeln nekar Itha AB:s filer med produktnamnet `PRIVERA` och version till och med 0.9.0.21. Den behövs eftersom 0.9.0-beta12 till 0.9.0-beta21 är giltigt signerade av Itha AB med Electrons spärrar avstängda (avsnitt 4.1), och den första regeln annars släpper in dem. I AppLocker går en nekande regel före en tillåtande. Den nekande regeln stoppar också `Privera.exe` i en äldre version som redan är installerad per dator under `C:\Program Files`, trots standardregeln där.

Den nekande regeln stoppar även installer och avinstallerare för 0.9.0-beta21 och äldre, eftersom de har samma produktnamn och versionen 0.9.0.0. En ny version tar bort den gamla med den gamla avinstalleraren. Uppgradera därför till 0.9.0-beta22 eller senare, eller avinstallera äldre versioner, innan regeln slås på.

Den nekande regeln gäller bara filer med produktnamnet `PRIVERA`. Motorn, OCR-motorn och `elevate.exe` i äldre versioner har inget produktnamn eller versionsnummer som skiljer dem från 0.9.0-beta22 och senare, och släpps in av den tillåtande regeln.

Reglerna som PowerShell, med `S-1-1-0` (Alla) eftersom namnet `Everyone` inte går att slå upp på en svensk Windows:

```
$tillat = [guid]::NewGuid()
$neka = [guid]::NewGuid()
@"
<AppLockerPolicy Version="1">
  <RuleCollection Type="Exe" EnforcementMode="NotConfigured">
    <FilePublisherRule Id="$tillat" Name="Privera: signerat av Itha AB" Description="Alla programfiler
signerade av Itha AB." UserOrGroupSid="S-1-1-0" Action="Allow">
      <Conditions>
        <FilePublisherCondition PublisherName="O=ITHA AB, L=SÖDERTÄLJE, S=STOCKHOLM, C=SE" ProductName="*"
BinaryName="*">
          <BinaryVersionRange LowSection="*" HighSection="*" />
        </FilePublisherCondition>
      </Conditions>
    </FilePublisherRule>
    <FilePublisherRule Id="$neka" Name="Privera: nekas till och med 0.9.0-beta21" Description="Privera.exe,
installer och avinstallerare med version till och med 0.9.0.21." UserOrGroupSid="S-1-1-0" Action="Deny">
      <Conditions>
        <FilePublisherCondition PublisherName="O=ITHA AB, L=SÖDERTÄLJE, S=STOCKHOLM, C=SE"
ProductName="PRIVERA" BinaryName="*">
          <BinaryVersionRange LowSection="*" HighSection="0.9.0.21" />
        </FilePublisherCondition>
      </Conditions>
    </FilePublisherRule>
  </RuleCollection>
</AppLockerPolicy>
"@ | Out-File -Encoding utf8 .\Privera-AppLocker.xml

# Pröva reglerna mot de installerade filerna innan de tillämpas.
$filer = (Get-ChildItem 'C:\Program Files\Privera' -Recurse -Filter *.exe).FullName
Test-AppLockerPolicy -XmlPolicy .\Privera-AppLocker.xml -Path $filer -User $env:USERNAME

# Lägg till reglerna i den lokala policyn. I en domän: importera filen i grupprincipen.
Set-AppLockerPolicy -XmlPolicy .\Privera-AppLocker.xml -Merge
```

Har organisationen redan en tillåtande regel för Itha AB, till exempel den som stod här före 0.9.0-beta22, räcker det att lägga till den nekande regeln.

Provat 2026-10-23 med `Test-AppLockerPolicy` och exakt den här policyn: `Privera.exe`, `sidecar.exe` och `tesseract.exe` ur ett signerat provbygge av 0.9.0-beta22, och installern för 0.9.0-beta22 och avinstalleraren i den, tillåts. `Privera.exe` och installern för 0.9.0-beta21, installern för 0.9.0-beta20 och avinstalleraren från en äldre version nekas av den nekande regeln, och ett program från en annan utgivare nekas eftersom ingen regel tillåter det. Inte provat: programmet under en tillämpad AppLocker-policy.

Använder organisationen AppLockers regler för DLL-filer behöver även filerna från Python Software Foundation, Microsoft och Intel tillåtas, och Itha AB:s egna DLL-filer, som har olika produktnamn och versioner och därför inte kan begränsas på version. Programmet startar Windows PowerShell för händelseloggen och för signaturkontrollen av uppdateringar. Händelseloggen påverkas av Constrained Language Mode, se Säkerhetsloggning avsnitt 7.3. Signaturkontrollen fungerar i Constrained Language Mode. Kan PowerShell inte startas alls, till exempel för att det är spärrat för användarna, installeras inga automatiska uppdateringar (avsnitt 5).

4.3 WDAC (App Control for Business)

Skapa reglerna från en dator där programmet är installerat per dator:

```
New-CIPolicy -FilePath .\Privera-WDAC.xml -ScanPath 'C:\Program Files\Privera' -Level Publisher -Fallback Hash
-UserPEs -MultiplePolicyFormat
```

Mot programfilerna i 0.9.0-beta20 gav kommandot signerregler för Itha AB, Python Software Foundation, Microsoft och Intel. Regeln för Itha AB binder till Microsofts mellancertifikat Microsoft ID Verified Code Signing PCA 2021 och namnet Itha AB, inte till det dagliga certifikatet, och bör därför hålla när certifikatet byts. Den godtar varje certifikat från samma mellancertifikat med namnet Itha AB. För en snävare regel kan signeraren dessutom bindas till signeringsprofilens egen EKU, som är densamma i alla Priveras certifikat: `1.3.6.1.4.1.311.97.811471209.363444190.628451136.88475081`. Microsoft beskriver det i dokumentationen för Trusted Signing (Artifact Signing).

En signerregel på nivån Publisher godtar alla versioner, också 0.9.0-beta12 till 0.9.0-beta21, där den signerade programfilen kan köra annan kod (avsnitt 4.1). Nivån FilePublisher binder regeln dessutom till filens namnuppgift och en lägsta filversion. Hur en sådan regel bör se ut för Privera, där `Privera.exe` saknar originalfilnamn, har vi inte provat. Inte provat under en tillämpad WDAC-policy.

5. Styrning av uppdateringar

Från version 0.9.0-beta21. Värdet `UpdateMode` (REG_DWORD) under `HKLM\Software\Policies\Privera`:

Värde	Läge	Vad programmet gör
0	Av	Frågar inte efter nya versioner. IT installerar nya versioner.
1	Kontrollera och meddela	Frågar licens.privera.se vid start och visar en rad om att en ny version finns och att organisationen installerar den. Hämtar och installerar ingenting.
2	Automatisk (förval)	Hämtar en ny version i bakgrunden och installerar den när användaren väljer det eller vid nästa normala avslut. Omstart tvingas aldrig fram.

```
reg add HKLM\Software\Policies\Privera /v UpdateMode /t REG_DWORD /d 0 /f
```

- Saknas värdet gäller 2. Policyn går före allt i programmet. Det finns ingen inställning där användaren kan ändra läget.
- Ett värde som inte går att tolka, av fel typ eller ett annat tal, ger 0 och räknas i säkerhetsloggens startpost som `ogiltiga_policyvarden`. Går registret inte att läsa vid start (tidsgräns fem sekunder) görs ingen kontroll den starten. Ett fel öppnar alltså aldrig för automatisk installation.
- Läget står i varje startpost i säkerhetsloggen: `app.start`, fältet `uppdateringar` (automatisk, kontrollera eller av).
- Användaren ser läget under Inställningar, Om Privera. När organisationen har satt policyn står det att uppdateringarna styrs av organisationen.
- Policyn läses vid start. En ändring gäller från nästa start.
- Oberoende av policyn görs ingen kontroll utan licensnyckel och ingen i avvecklingsläget.
- Versioner före 0.9.0-beta21 bortser från värdet och uppdaterar automatiskt.

Äkthetskontrollen (från 0.9.0-beta22). Innan en hämtad uppdatering installeras kontrollerar programmet den med Windows PowerShell. Godkänt är bara att Authenticode-signaturen har status `Valid`, att den gäller den hämtade filen och att utgivarens namn har alla delarna `CN=Itha AB, O=Itha AB, L=Södertälje, S=Stockholm, C=SE` med exakt de värdena. Därefter ska filens versionsuppgift vara den version som uppdateringsmanifestet angav, så att en äldre installer med giltig signatur inte kan installeras som en nyare. En fil vars versionsuppgift saknas, inte går att läsa eller bara är nollor avvisas. Paket för webbinstallation godtas inte. PowerShell startas med full sökväg i Windows-katalogen och med en egen, minimal miljö. Allt som inte är ett uttryckligt godkännande är ett nej: uppdateringen installeras inte, och nejet loggas som `uppdatering.signatur_nekad` (5003). Kontrollen är provad mot den signerade installern för 0.9.0-beta21, som godkändes med rätt utgivare och fälldes med ett annat utgivarnamn, efter en ändrad byte och som osignerad.

När PowerShell är spärrat. Kan kontrollen inte köras, till exempel för att PowerShell är spärrat för användarna, installeras inga automatiska uppdateringar. Bara när PowerShell inte går att starta sparas nejet i programmets datamapp, och nästa försök görs då först efter 24 timmar, så att installern inte hämtas om vid varje start. Vid

tidsgräns, fel utgivare, ogiltig signatur eller fel version nekas uppdateringen utan att något sparas, och nästa start försöker igen. Under Inställningar, Om Privera står då att uppdateringarna inte kunde verifieras på datorn, att inga uppdateringar installeras och att användaren ska kontakta IT-avdelningen. Sätt i så fall `UpdateMode` till 0 och rulla ut nya versioner centralt.

Före 0.9.0-beta22 jämförde utgivarkontrollen bara certifikatets namn (CN), och där PowerShell inte kunde köras hoppades signaturkontrollen över. En dator som uppdaterar från 0.9.0-beta21 till 0.9.0-beta22 prövar installern med den äldre kontrollen. Från 0.9.0-beta22 gäller den nya.

Installation per dator. Programmappen under Program Files är skrivskyddad för vanliga användare. En automatisk uppdatering startar installern, som då begär förhöjning (UAC). En användare utan administratörsrätt kan inte godkänna den, och uppdateringen blir inte av. Sätt därför `UpdateMode` till 0 eller 1 vid installation per dator. Det följer av installerns mall och är inte provat.

Säkerhetsuppdateringar. Rättningar ges bara i den senaste versionen, se Livscykel och support. Med läget Av eller Kontrollera och meddela ansvarar organisationen för att rulla ut dem.

6. Registerpolicyer och ADMX-mall

Programmet läser de här värdena under `HKLM\Software\Policies\Privera` vid start, med 64-bitarsvyn. Nyckeln är bara skrivbar för administratörer. Inga andra värden läses. Värdena läses med `reg.exe`. Från 0.9.0-beta22 startas det med full sökväg i Windows-katalogen, som programmet tar ur den plats som Windows laddade systembiblioteket `ntdll.dll` från, inte ur miljövariabeln `SystemRoot`, och med en egen, minimal miljö. Tidigare versioner tog katalogen ur `SystemRoot`. Policyn är ändå en styrning för vanlig användning, inte ett skydd mot en användare som avsiktligt vill kringgå den, se Säkerhetsloggning avsnitt 7.1.

Värde	Typ	Förval	Betydelse	Från version
`SecurityLogEnabled`	REG_DWORD	1	0 = ingen loggfil.	0.9.0-beta20
`SecurityLogDirectory`	REG_SZ eller REG_EXPAND_SZ	programmets datamapp	Katalog för säkerhetsloggens fil. Absolut lokal sökväg eller UNC, högst 240 tecken, utan `..`.	0.9.0-beta20
`EventLogEnabled`	REG_DWORD	1	0 = inget till Windows händelselogg.	0.9.0-beta20
`AllowClearExportLogs`	REG_DWORD	1	0 = "Rensa mappen" i Historik nekas och loggas.	0.9.0-beta20
`RequireEventLogForExport`	REG_DWORD	0	1 = en export vägras om den inte kunde registreras i Windows händelselogg (kräver `EventLogEnabled` = 1). På en dator där händelseloggen inte fungerar, till exempel för att PowerShell är spärrat, vägras då alla exporter. Varje export försöker skriva till händelseloggen igen och väntar upp till cirka 10 sekunder om PowerShell hänger. Pröva att händelseloggen fungerar innan värdet sätts.	0.9.0-beta22
`UpdateMode`	REG_DWORD	2	0 = av, 1 = kontrollera och meddela, 2 = automatisk (avsnitt 5).	0.9.0-beta21
`AllowDebugPort`	REG_DWORD	0	1 = kundutgåvan får startas med en felsökningsport (`--remote-debugging-port` med flera, avsnitt 4.1). Används bara av Itha AB:s sandlådetest. Loggas som 1006.	0.9.0-beta22

Loggvärdena beskrivs i Säkerhetsloggning avsnitt 7.

ADMX-mallen. Privera.admx med språkfilerna sv-SE/Privera.adml och en-US/Privera.adml. Principerna hamnar under Datorkonfiguration, Administrativa mallar, Privera. Mallen beror inte på någon annan ADMX-fil.

- Grupprincip: lägg `Privera.admx` i `PolicyDefinitions`, lokalt `C:\Windows\PolicyDefinitions` eller i domänens centrala arkiv under `SYSVOL`, och språkfilerna i undermapparna `sv-SE` och `en-US`.
- Intune: Enheter, Konfiguration, Importera ADMX. Ladda upp `Privera.admx` med `en-US\Privera.adml`. Intune godtar bara en språkfil per mall och bara en-US. Skapa sedan en profil av typen Importerade administrativa mallar. En ny version av mallen godtas inte förrän profilerna och den gamla mallen är borttagna.
- Inte konfigurerad ger förvalet för alla principer. För de fem av- och på-principerna skriver Inaktiverad värdet 0. För katalogen och uppdateringarna tar Inaktiverad bort värdet, så att förvalet gäller.
- Mallen kontrolleras mot programmet i ett automatiskt test: varje värde mallen skriver läses av programmet med samma namn, typ och tal, och varje värde programmet läser finns i mallen. Den är inte provad i grupprincipredigeraren eller i Intune.

7. Nätverk

Programmet öppnar ingen inkommande port. Den utgående trafiken, till licens.privera.se och, första gången språkmodellen behövs, till Hugging Face, står i tabellen i Underlag för IT och informationssäkerhet. Med `UpdateMode` = 0 försvinner uppdateringskontrollen. Licenstrafiken finns kvar: aktivering, förnyelse normalt var sjunde dag och högst 14 dagar utan kontakt med licensservern.

8. VDI och terminalserver

Läget i dag, enligt programmets kod:

- Licensen binds till ett maskin-id som räknas fram ur ett slumpat installations-id i Windows-kontots profil (`%APPDATA%\privera-desktop`) och datornamnet. Varje Windows-konto får därför ett eget maskin-id, även på samma dator, och en licens gäller ett maskin-id åt gången.
- Terminalserver med flera sessioner: varje användare behöver en egen licens. Varje session kör en egen motor med språkmodellen. Minnesbehovet per session är inte mätt.
- Beständiga virtuella skrivbord med samma datornamn och en profil som behålls fungerar som en vanlig dator.
- Poolade skrivbord som byter datornamn eller inte behåller profilen ger ett nytt maskin-id. Licensen måste då tas över, och en övertagning går högst en gång var 30:e dag. Det fungerar inte i praktiken.
- Språkmodellen, cirka 500 MB, hämtas till varje användares profil första gången den behövs. Med profiler som inte behålls hämtas den igen.
- Licenser per organisation, per dator eller för terminalserver finns inte i dag. Kontakta support@privera.se.

Inget av detta är provat i en VDI- eller terminalservermiljö.

9. Det som är provat och det som inte är det

Provat i Windows Sandbox 2026-10-07 (ren Windows 11 utan Python, Node, Git eller Visual C++-runtime): tyst installation per användare (`/S`) av 0.9.0-beta20, uppgradering till 0.9.0-beta21 med `/S` ovanpå den, att licensen och licensfilen överlever uppgraderingen, att alla programfiler är giltigt signerade, att Appar och funktioner visar utgivaren Itha AB, analys med OCR och språkmodell, och `UpdateMode` = 1 under `HKLM\Software\Policies\Privera` i det installerade programmet.

Provat 2026-10-23 med `Test-AppLockerPolicy` mot ett signerat provbygge av 0.9.0-beta22: AppLocker-regeln i avsnitt 4.2, med resultatet som står där.

Inte provat:

- Tyst installation, uppgradering och avinstallation i Intune och Configuration Manager.
- Installation per dator på en ren dator, och automatisk uppdatering av en sådan installation.
- ADMX-mallen i grupprincipredigeraren och i Intune.
- Programmet under en tillämpad AppLocker- eller WDAC-policy.

- VDI och terminalserver.
- Värdena 0 och 2 för `UpdateMode` i ett installerat program (provade i automatiska tester).
- Uppdateringen på en dator där PowerShell är spärrat, i ett installerat program (provad i automatiska tester).
- `RequireEventLogForExport` och `AllowDebugPort` i ett installerat program (provade i automatiska tester).

Frågor om utrullning: support@privera.se.

Sårbarhetshantering

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/sarbarhetshantering/

Gäller Windows-programmet Privera, licensservern licens.privera.se och webbplatsen privera.se. Den som vill rapportera en sårbarhet hittar samma regler i kort form på privera.se/security/sarbarheter/. Incidenter och rapportering till myndigheter beskrivs i Incidentrutin.

1. Läget i korthet

Del	Läge
Kontaktväg	feedback@privera.se. Bevakas helgfria vardagar.
Bemanning	En person hanterar säkerhetsärenden. Ingen ersättare är utsedd, och det finns ingen jour på kvällar och helger.
security.txt (RFC 9116)	privera.se/.well-known/security.txt
Bedömning med CVSS och tidsramar för åtgärd	Avsnitt 3 och 4. Tidsramarna är fastställda 2026-10-23.
Samordnat röjande	Avsnitt 6.
Säkerhetsmeddelanden till kunder	Rutin i avsnitt 7. De två första meddelandena, PRIVERA-SA-2026-001 och PRIVERA-SA-2026-002, publicerades 2026-11-12 med 0.9.0-beta23.
Rapportering enligt EU:s cyberresiliensförordning (CRA), artikel 14	Rutin finns i Incidentrutin. Enligt ENISA:s anvisning registrerar sig tillverkaren i plattformen när en anmälan ska göras, inte i förväg. Registreringen tar några minuter och anmälan kan lämnas direkt; CERT-SE prövar behörigheten parallellt. Inloggningen som krävs (EU Login med tvåfaktorautentisering) finns sedan 2026-10-07.
Automatisk bevakning av sårbarheter i beroenden	Delvis. Larm från GitHub (Dependabot) för npm-beroendena sedan 2026-10-23. Larmen ger inga automatiska rättningar och körs inte i CI. Motorns Python-paket och OCR-motorns bibliotek bevakas manuellt.
Komponentförteckning (SBOM)	Publiceras för 0.9.0-beta23 i CycloneDX 1.5. Från 0.9.0-beta22 tas den fram ur det byggda paketet i bygget. Har kända luckor, bland annat ingen sårbarhetsskanning mot den. Se SBOM.
PGP-nyckel för krypterade rapporter	Ej infört.
Maskinläsbara säkerhetsmeddelanden (CSAF)	Ej infört.

2. Mottagning

Rapporter tas emot på feedback@privera.se. Telefon 076 926 35 70 kan användas för att påskynda ett ärende men besvaras i mån av möjlighet.

Rapporter som kommer in via support@privera.se eller info@privera.se, via en kund eller via en leverantör (till exempel en säkerhetsnotis om Electron eller Tesseract) förs över till samma ärendegång.

Vid mottagning:

1. Mottagandet bekräftas till rapportören inom 2 arbetsdagar.
2. Ärendet får ett id och en ärendelogg: mottaget, bedömning, beslut, åtgärd, version och kommunikation. Verkliga personuppgifter och kunddokument läggs aldrig i loggen.
3. Innehåller rapporten verkliga dokument ber vi om en reproduktion med påhittade uppgifter. Itha AB tar inte emot kunders dokument i ett säkerhetsärende, utom efter ett särskilt personuppgiftsbiträdesavtal för felsökning, se Underleverantörer avsnitt 1.

Inkorgen bevakas helgfria vardagar. Det finns ingen jour dygnet runt, och säkerhetsärenden hanteras i dag av en person.

3. Bedömning

Första bedömning görs inom 5 arbetsdagar från mottagandet. Bedömningen svarar på:

- Går felet att återskapa i den senaste versionen? Bara den senaste versionen lämnas ut i uppdateringskanalen.
- Vilka delar berörs: programmet, uppdateringskedjan, licensservern eller webben?
- Är sårbarheten aktivt utnyttjad? Då gäller rapporteringen enligt CRA artikel 14 direkt, oavsett poäng.
- CVSS-poäng, med CVSS v4.0 eller v3.1 om rapportören angett den. Poäng och vektor skrivs in i ärendeloggen.

Allvarlighetsgrader

Grad	CVSS	Exempel för Privera
Kritisk	9,0 till 10,0	Kodkörning via en öppnad PDF- eller Word-fil. En manipulerad uppdatering som installeras. En uppgift som programmet visar som maskerad finns synlig i den exporterade filen.
Hög	7,0 till 8,9	En uppgift som visas som maskerad finns kvar dold i utfilen (textlager, metadata, bildlager) utan varning. Åtkomst till andra kunders licensuppgifter.
Medel	4,0 till 6,9	Begränsat informationsläckage från licensservern. Fel som kräver lokal åtkomst till användarens konto.
Låg	0,1 till 3,9	Härdningsbrister utan känd utnyttjandeväg.

En exporterad fil som innehåller en uppgift som programmet har markerat som maskerad räknas alltid som minst Hög, även om CVSS-poängen blir lägre. Det är den skada produkten finns för att förhindra.

Det här är inte en sårbarhet: att den automatiska detektionen missar en uppgift som användaren aldrig markerade. Detektionen hittar inte allt, och dokumentet ska alltid granskas före export. Sådana fel hanteras som kvalitetsfel.

4. Tidsramar för åtgärd

Tidsramarna är fastställda 2026-10-23. De räknas från att bedömningen är klar. Åtgärd betyder en rättad version i uppdateringskanalen eller, om det inte går i tid, en publicerad skyddsåtgärd som minskar risken.

Grad	Åtgärd senast
Kritisk	7 dagar
Hög	30 dagar
Medel	90 dagar
Låg	Nästa planerade version, senast 180 dagar
Aktivt utnyttjad	Snarast. Råd om skyddsåtgärd till berörda kunder senast i uppdateringen 72 timmar efter kännedom, räknat under helgfria vardagar, se Incidentrutin avsnitt 5.

Kan en tidsram inte hållas meddelas rapportören och, för Kritisk och Hög, berörda kunder med skäl och ny tidpunkt.

Sårbarheter i tredjepartskomponenter (Electron och Chromium, Python-paket, Tesseract med bibliotek, språkmodellen) bedöms på samma sätt, efter hur de kan nås i Privera.

Före varje version med en säkerhetsrättning körs de ordinarie releasekontrollerna, och ett regressionstest för felet läggs till.

5. Leverans av rättningar

- Rättningar levereras som en ny version i den ordinarie uppdateringskanalen. Programmet hämtar uppdateringen i bakgrunden och installerar den när användaren väljer det eller vid nästa avslut.
- Uppdateringar lämnas bara ut till licenser som har rätt att köra programmet: provperiod, aktivt abonnemang, uppsagt abonnemang med betald period kvar och misslyckad kortdragning medan betalningen försöks igen. I

samtliga fall gäller det till periodens slut plus 3 dagar. Uppsagda, obetalda, pausade och utgångna licenser får inga uppdateringar, inte heller säkerhetsrättningar. Programmet bearbetar inga dokument utan giltig licens.

- Undantag: när Itha AB har publicerat ett avvecklingsbesked lämnas slutversionen ut till varje känd licens, också uppsagda. Se Avvecklingsläge.
- Installer och uppdateringar är signerade av Itha AB. Programmet kontrollerar SHA-512 och signaturen före installation. Från 0.9.0-beta22 kräver kontrollen Itha AB:s hela utgivarnamn och den version som uppdateringsmanifestet angav, och den är stängd vid fel: kan äktheten inte kontrolleras, till exempel för att Windows PowerShell är spärrat, installeras uppdateringen inte, och programmet säger det under Om Privera. Se IT-drift avsnitt 5. I 0.9.0-beta21 och äldre jämförde kontrollen bara certifikatets namn (CN) och hoppades över där PowerShell inte kunde köras.
- Kunder som blockerar automatiska uppdateringar eller sällan är uppkopplade kan hämta den nya installern på privera.se/download/ med sin licensnyckel, under samma villkor som ovan.
- Det finns en uppdateringskanal. Separata säkerhetsuppdateringar utan funktionsändringar finns inte i dag. När en rättning inte kan vänta på nästa planerade version ges en separat version ut.
- I avvecklingsläge hämtar programmet inga uppdateringar. Rättningar efter en avveckling kan bara nå kunden som en ny installer: via privera.se/download/ så länge licensservern finns, och därefter med e-post eller på annat sätt som Itha AB eller en konkursförvaltare ordnar.
- Itha AB kan inte se vilken version en kund kör. Licensservern registrerar vilka licenser som har hämtat en viss installationsfil, men inte vilken version som körs.

Säkerhetsuppdateringar ingår i abonnemanget utan extra kostnad. Supportperioden beskrivs i Livscykel och support.

6. Samordnat röjande

- Rapportören får besked om bedömningen och om planerad tidpunkt för rättning.
- Vi publicerar inga detaljer förrän en rättning finns eller 90 dagar har gått från mottagandet, beroende på vad som kommer först. Tidpunkten kan ändras i samråd med rapportören, till exempel om en rättning är nära eller om sårbarheten redan är känd.
- Rapportören nämns i säkerhetsmeddelandet om hen vill.
- Det finns inget belöningsprogram.
- Den som i god tro söker och rapporterar sårbarheter enligt policyn polisanmäls inte av Itha AB. Policyn tillåter inte test mot andra kunders licenser eller data, överbelastning av licens.privera.se, social manipulation eller test med verkliga personuppgifter.
- Berör en sårbarhet en tredjepartskomponent meddelar vi även komponentens förvaltare, enligt CRA artikel 13.6.
- Itha AB är inte CVE-utfärdare (CNA). Vid behov begärs CVE via MITRE. Ingen CVE har begärts hittills.

7. Säkerhetsmeddelanden

Ett säkerhetsmeddelande ges ut för varje rättad sårbarhet av grad Medel eller högre och för varje aktivt utnyttjad sårbarhet.

Meddelandet innehåller id, datum, berörda versioner, rättad version, allvarlighetsgrad och CVSS-vektor, en kort beskrivning utan utnyttjandedetaljer före rättning, vad kunden ska göra (till exempel installera en viss version eller kontrollera tidigare exporterade filer) och tack till rapportören.

Kanaler:

1. privera.se/security/sarbarheter/, listan över publicerade säkerhetsmeddelanden.
2. E-post till den adress som angavs vid köpet och till kontaktpersonen i avtalet, när ett avtal finns.

Andra kanaler finns inte. Programmet visar inga säkerhetsmeddelanden, och det finns inga publicerade versionsanteckningar för de aktuella versionerna.

Kan en sårbarhet ha gjort att tidigare exporterade filer innehåller kvarvarande personuppgifter säger meddelandet det tydligt och beskriver hur kunden kan kontrollera filerna. Kunden är personuppgiftsansvarig för dokumenten och avgör själv om det är en personuppgiftsincident hos kunden.

8. Rapportering enligt CRA

Sedan 2026-09-11 ska tillverkare rapportera aktivt utnyttjade sårbarheter och allvarliga incidenter som påverkar produktens säkerhet, enligt förordning (EU) 2024/2847 artikel 14: tidig varning inom 24 timmar, anmälan inom 72 timmar och slutrapport senast 14 dagar efter att en rättning eller skyddsåtgärd finns. För allvarliga incidenter lämnas slutrapporten inom en månad efter anmälan. Rapporten lämnas i ENISA:s gemensamma rapporteringsplattform och går till CERT-SE. Berörda användare ska informeras.

Fristerna gäller alla dagar i veckan. Inkorgen bevakas helgfria vardagar och det finns ingen jour eller ersättare, så Itha AB kan inte garantera att en rapport som kommer in på en helg eller helgdag blir läst och rapporterad inom 24 timmar.

Hur det görs steg för steg står i Incidentrutin.

9. Det som återstår

- En utsedd ersättare för säkerhetsärenden, och bevakning av inkorgen även på helger.
- Automatisk bevakning av sårbarheter i beroenden i CI (npm audit, pip-audit, Dependabot), också för motorns Python-paket och OCR-motorns bibliotek.
- Sårbarhetsskanning mot SBOM:en, och licensservern i SBOM:en.
- Övning av incidentrutinen.

Incidentrutin

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/incidentrutin/

Beskriver hur Itha AB hanterar säkerhetsincidenter i Privera och rapporterar dem till myndigheter och kunder. Rutinen beskriver hur vi arbetar och är inte i sig ett avtal. Frister blir avtalsvillkor först när en kund har undertecknat säkerhetsbilagan till avtalet. Bilagan finns som utkast för juristgranskning, och ingen kund har undertecknat den per 2026-10-23. Hör ihop med Sårbarhetshantering.

1. Roller

Itha AB är ett litet bolag, och rollerna är få.

Roll	Uppgift	Läge
Incidentansvarig	Bedömer om något är en incident, beslutar åtgärder, rapporterar till CERT-SE, ENISA och IMY och informerar kunder.	En person.
Ersättare	Samma befogenheter när den incidentansvarige inte nås inom 12 timmar.	Inte utsedd.
Teknisk åtgärd	Stoppar spridning, rättar, bygger och signerar en ny version.	Den incidentansvarige, med stöd av den som har behörighet till berört system.

Så länge ingen ersättare finns kan en frist på 24 timmar inte garanteras när den incidentansvarige inte är nåbar. Det är rutinens största svaghet och ska åtgärdas först.

Kontaktuppgifter:

Vad	Var
In till Itha AB	feedback@privera.se, telefon 076 926 35 70
Dataskyddsfrågor	info@privera.se
CERT-SE	cert@cert.se, 010-382 80 00, dygnet runt
CRA-rapportering	ENISA:s gemensamma rapporteringsplattform. Enligt ENISA:s anvisning registrerar sig tillverkaren i plattformen när en anmälan ska göras, inte i förväg. Registreringen tar några minuter och anmälan kan lämnas direkt; CERT-SE prövar behörigheten parallellt.
IMY (personuppgiftsincident)	Anmälan via imy.se

2. Vad som är en incident

Följande startar rutinen:

1. Aktivt utnyttjad sårbarhet i Privera, i uppdateringskedjan eller i en ingående komponent (CRA artikel 14.1).
2. Allvarlig incident som påverkar produktens säkerhet: den påverkar eller kan påverka förmågan att skydda tillgänglighet, äkthet, riktighet eller konfidentialitet hos känsliga eller viktiga data eller funktioner, eller har lett eller kan leda till att skadlig kod förs in eller körs (CRA artikel 14.5). Exempel: signeringen eller uppdateringskanalen har missbrukats.
3. Personuppgiftsincident i Itha AB:s egna uppgifter: licensdatabasen, betaltjänsten eller e-posten.
4. Kvarvarande uppgift i en exporterad fil som en kund upptäcker: en uppgift som programmet visade som maskerad finns kvar i utfilen. Det är den incidenttyp produkten främst riskerar. Den hanteras alltid som en sårbarhet och blir en CRA-incident om den utnyttjas.

Klockan startar när Itha AB får kännedom. Inkorgen bevakas helgfria vardagar och det finns ingen jour. Ett meddelande som kommer en fredagskväll kan därför bli läst först på måndagen. Planerat: mobilavisering för inkorgen, så att den incidentansvarige ser meddelandet samma dag.

3. Förlopp

Steg 1. Ta emot och registrera (genast)

Ett ärende öppnas och tidpunkten för kännedom skrivs ner. Den styr alla frister.

Steg 2. Bedöm (inom några timmar)

Är det en av typerna i avsnitt 2? Är det aktivt utnyttjat? Vilka kunder kan beröras? Itha AB ser inte vilken version en kund kör, bara vilka licenser som har hämtat en viss installationsfil.

Steg 3. Begränsa

Åtgärderna beror på fallet:

- Stoppa uppdateringar och nedladdning genom att ta bort den berörda versionen ur både uppdateringskanalen och nedladdningen. Det räcker inte att ta bort den ur uppdateringskanalens manifest.
- Redan nedladdade uppdateringar kan inte återkallas från servern. Programmet hämtar en ny version i bakgrunden och installerar den vid nästa avslut. Kan en skadlig version redan ha hämtats varnas kunderna per e-post och telefon, snarast och senast inom fristerna i avsnitt 5.
- Om signeringen har missbrukats: stoppa byggen, byt åtkomsten till signeringstjänsten och begär att Microsoft återkallar berörda certifikat. Förfarandet är inte prövat.
- Om licensserverns databas berörs: byt licensserverns hemligheter och betaltjänstens nycklar och granska de åtkomstloggar som finns hos leverantörerna. Licensserverns egen säkerhetslogg sparas inte i dag, se Säkerhetsloggning avsnitt 11.
- Vid kvarvarande uppgifter i exporten: ge kunderna en skyddsåtgärd, till exempel att inte använda en viss exportväg eller filtyp, tills en rättad version finns.

Steg 4. Rapportera

Se avsnitt 4 och 5. Frister räknas från kännedom.

Steg 5. Rätta och följ upp

Rätta enligt tidsramarna i Sårbarhetshantering, ge ut säkerhetsmeddelande och lämna slutrapport. Avsluta med en kort genomgång: vad hände, vad tog tid och vad ändras i rutinen.

4. Rapportering till myndigheter

Händelse	Mottagare	Frister
Aktivt utnyttjad sårbarhet (CRA artikel 14.1 och 14.2)	ENISA:s rapporteringsplattform, som förmedlar till CERT-SE	Tidig varning inom 24 timmar. Anmälan inom 72 timmar. Slutrapport senast 14 dagar efter att rättning eller skyddsåtgärd finns.
Allvarlig incident (CRA artikel 14.3 och 14.4)	Samma	Tidig varning inom 24 timmar. Anmälan inom 72 timmar. Slutrapport inom en månad efter anmälan.
Personuppgiftsincident i köp-, licens- eller supportuppgifter (GDPR artikel 33)	IMY	Inom 72 timmar, utom när det är osannolikt att incidenten medför en risk för de registrerade. Berörda kunder underrättas också.

Innehåll enligt CRA artikel 14.2: den tidiga varningen anger, om det är känt, i vilka medlemsstater produkten har tillhandahållits (i dag Sverige). Anmälan efter 72 timmar beskriver produkten, sårbarhetens och utnyttjandets art, vidtagna åtgärder och vad användarna själva kan göra. Slutrapporten beskriver allvarlighet och påverkan, vad som är känt om angriparen och vilken säkerhetsuppdatering som rättar felet.

5. Vad kunden får

Berörda kunder underrättas utan onödigt dröjsmål, och senast enligt tabellen. Fristerna är satta efter vad en person utan jour kan hålla. Samma frister för tidig varning och uppdatering står i utkastet till säkerhetsbilaga, som också har frister för slutrapporten.

Steg	Senast
Tidig varning, kännedom en helgfri vardag	24 timmar efter kännedom, räknat under helgfria vardagar.
Tidig varning, kännedom en lördag, söndag, helgdag, midsommarafton, julafton eller nyårsafton	Klockan 12 närmast följande helgfria vardag.
Uppdatering, med råd om skyddsåtgärd när en sådan finns	72 timmar efter kännedom, räknat under helgfria vardagar.

Räknat under helgfria vardagar betyder att tid på lördagar, söndagar, helgdagar, midsommarafton, julafton och nyårsafton inte räknas in. Exempel: kännedom en fredag klockan 17 ger tidig varning senast klockan 17 närmast följande helgfria vardag. Kännedom en lördag ger tidig varning senast klockan 12 närmast följande helgfria vardag.

När kännedomen uppkommer nära eller under en helg kan kunden alltså få den tidiga varningen senare än de 24 timmar som gäller för kundens egen tidiga varning enligt cybersäkerhetslagen (NIS2). Itha AB:s egen rapportering enligt CRA artikel 14 (avsnitt 4) har en frist på 24 timmar som gäller alla dagar och påverkas inte av tabellen.

Underrättelsen ges i steg som passar kundens egen rapportering enligt cybersäkerhetslagen:

Steg	Innehåll
Tidig varning	Vad som har hänt så långt det är känt, om kunden kan vara berörd, omedelbara skyddsåtgärder och kontaktperson hos Itha AB.
Uppdatering	Första bedömning av orsak och omfattning, berörda versioner, åtgärder och vad kunden ska göra.
Slutrapport	Orsak, påverkan, rättad version och åtgärder för att det inte ska hända igen.
Under hela förloppet	Svar på kundens frågor och det underlag kunden behöver för sin egen anmälan till CERT-SE, IMY eller annan myndighet.

Inkorgen bevakas bara helgfria vardagar, och det finns ingen jour och ingen ersättare. Därför räknas fristen ovan under helgfria vardagar.

Vilka kunder som informeras: alla som kan beröras. Eftersom Itha AB inte ser vilken version en kund kör informeras i tveksamma fall alla med aktiv licens.

Kontaktväg ut: e-post till kundens registrerade adress och till kontaktpersonen i avtalet, när ett avtal finns. Kritiska ärenden även per telefon till kontaktpersonen, om numret finns. Därtill ett säkerhetsmeddelande på privera.se/security/sarbarheter/.

Kontaktväg in: feedback@privera.se, telefon 076 926 35 70.

När kunden själv hittar en kvarvarande uppgift i en fil: dokumenten finns bara hos kunden, och Itha AB ser dem inte. Kunden är personuppgiftsansvarig och avgör om det är en personuppgiftsincident hos kunden. Itha AB hjälper till att fastställa orsaken. Beskriv felet eller återskapa det med påhittade uppgifter, och skicka inte det verkliga dokumentet.

6. Läge

Punkt	Läge
Kontaktadresser	Klart. Säkerhet till feedback@privera.se, dataskyddsfrågor till info@privera.se.
Ersättare	Inte utsedd.

Punkt	Läge
EU Login med tvåfaktorautentisering (krävs för att registrera sig och anmäla)	Klart 2026-10-07. Registrering i plattformen görs vid första anmälan, enligt ENISA:s anvisning.
Mobilavisering för inkorgen	Inte gjord.
Säkerhetsbilaga till avtal (incidenter, revision och insyn)	Utkast för juristgranskning, 2026-10-23. Ingen kund har undertecknat den. Lämnas ut på begäran till info@privera.se.
Övning av rutinen (ett scenario: signeringen missbrukad, uppdateringskanalen manipulerad eller kvarvarande uppgift hos en kund)	Inte genomförd.
Kontinuitet om Itha AB upphör	Kunderna får en slutversion som fungerar utan licensserver. Mekanismen finns från 0.9.0-beta20, se Avvecklingsläge.

Säkerhetsloggning

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/loggning/

Beskriver säkerhetsloggen i Privera: vad som loggas, hur loggen skyddas mot ändring, hur organisationen styr den och hur den samlas in till SIEM. Säkerhetsloggen finns från version 0.9.0-beta20. Äldre versioner har bara exportloggen (avsnitt 9) och loggar inte stoppade exportförsök eller andra säkerhetshändelser. Dokumentet beskriver 0.9.0-beta22. Det som är nytt i den versionen står utskrivet. Licensserverns logg beskrivs i avsnitt 11.

1. Sammanfattning

Del	Läge
Säkerhetshändelser loggas, inte bara exporter	Infört. 31 händelsekoder från 0.9.0-beta23, 30 i 0.9.0-beta22: start, licens, avveckling, analys, export, läckkontroll, rensning, inställningar, uppdatering och loggens egen hälsa (avsnitt 5). Analys och påbörjad export loggas från 0.9.0-beta22.
Exportförsök som stoppas	Infört, även stopp innan motorn startar: licens, sökväg, saknat eller ändrat fingeravtryck, oläsbar källfil och vald utfil som är källfilen.
Exporten i händelseloggen innan utfilen skrivs	Från 0.9.0-beta22. Exporten väntar på kvitto från Windows händelselogg innan motorn skriver utfilen, högst tio sekunder. Med policyn `RequireEventLogForExport` vägras en export som inte fick kvitto (avsnitt 2).
Manipulationsskydd	Delvis. Programmet kan upptäcka manipulation av loggen men inte hindra den. Det som kan hindra den är organisationens egna åtgärder: append-only-behörighet på loggkatalogen (avsnitt 7.2) och insamling till SIEM. Hashkedja (SHA-256) och ankare upptäcker ändrade, borttagna och omordnade poster, kortad eller raderad fil och rader som lagts till utanför programmet, så länge ankaret är orört. Kedjan har ingen hemlig nyckel: den som kan skriva om både filen och ankaret, alltså användaren själv, kan dölja ändringar lokalt. Kopian i Windows händelselogg skyddar bara delvis, eftersom en vanlig användare kan fylla loggen så att äldre poster skrivs över och en administratör kan rensa den. Det som organisationens SIEM redan har samlat in kan användaren inte ändra. En post som användaren själv skriver både i filen och i händelseloggen går inte att skilja från programmets (avsnitt 4, 7.1 och 10).
Licensservern	Delvis. Skriver en säkerhetsrad per händelse, med pseudonymer i stället för licensnyckel och IP-adress, men raderna sparas inte och når inte organisationens SIEM (avsnitt 11).
Integration med SIEM	Delvis. Windows händelselogg (Application, källan `Privera`) med fasta händelse-ID, och en JSONL-fil i en katalog som organisationen kan styra med policy. Under AppLocker eller WDAC (PowerShell Constrained Language Mode) krävs registrerad händelsekälla (avsnitt 7.3). Insamlingen är inte provad med Windows Event Forwarding eller i något SIEM (avsnitt 8).
Central styrning	Infört. Registerpolicy under `HKLM\Software\Policies\Privera`: loggkatalog, fillogg på eller av, händelselogg på eller av och förbud mot att rensa exportloggar. Går policyn inte att läsa nekas rensning. Från 0.9.0-beta21 också uppdateringarna, och från 0.9.0-beta22 krav på kvitto från händelseloggen före export. En ADMX-mall för alla värden finns, se IT-drift avsnitt 6. Policyn styr vanlig användning men skyddar inte mot en användare som avsiktligt vill kringgå den (avsnitt 7.1).
Inga personuppgifter eller dokumentinnehåll	Infört. Fälten är typade och fritext kommer inte in (avsnitt 6). Undantag: Windows-kontots namn och datornamnet står i varje post, eftersom de visar vem som gjorde vad.
Exportloggen (`.audit.json`)	Finns kvar som förut, med ett nytt fält för stoppkod (avsnitt 9).

2. Var loggen skrivs

Fil. En fil per dator och Windows-konto:

```
privera-sakerhetslogg-<DATORNAMN>-<konto>.jsonl
```

Tecken utanför `A-Z a-z 0-9 . _ -` blir `_`. Förvald katalog är programmets datamapp, `%APPDATA%\privera-desktop\sakerhetslogg\`. Med policyn `SecurityLogDirectory` skrivs filen i stället i den katalog organisationen anger, lokal eller UNC. Flera datorer kan dela en katalog, eftersom varje dator och konto har sin egen fil och sin egen kedja.

Går policyns katalog inte att skriva till (den saknas, nätet är nere eller behörighet saknas) skrivs loggen i den förvalda katalogen, och den första posten där är `logg.skrivfel` (fel). Den går också till händelseloggen. Filen i reservkatalogen har samma namn men en egen kedja som börjar på `seq` 1. När policyns katalog nås igen fortsätter policyfilens kedja där den slutade, och filen prövas mot sitt ankare (avsnitt 4).

Ankare. I programmets datamapp ligger ett ankare per dator och konto, `sakerhetslogg-ankare-<DATORNAMN>-<konto>.json`. Det minns, för varje loggfil som datorn och kontot har använt, löpnummer och hash för den senast skrivna posten, filens längd i byte och SHA-256 av hela filen efter programmets senaste skrivning, samt den senast använda filen och programversionen. Från 0.9.0-beta23 minns det också vilka poster i filen Windows händelselogg ännu inte har kvitterat. Med roamande `%APPDATA%` skriver varje dator bara sitt eget ankare. Ankaret ligger aldrig i policyns katalog.

Loggfilen identifieras av sin verkliga sökväg på disk: junctions, symboliska länkar, `subst` och korta 8.3-namn löses upp, så att en annan stavning av samma katalog inte blir en annan fil. Nätverkssökvägar (UNC, mappade enheter) lämnas som de anges.

Windows händelselogg. Loggen *Application*, källan `Privera`. Varje post i filen skrivs också dit, med samma rad och radens egen hash. Posterna samlas i upp till 750 millisekunder och skrivs sedan i ett anrop till Windows PowerShell. Ett anrop räknas som lyckat först när PowerShell har kvitterat det. Från 0.9.0-beta23 bär kvittot antalet poster som skrevs, och programmet godtar det bara när varje post i anropet är skriven. Annars räknas anropet som misslyckat och görs om.

Poster som inte nådde händelseloggen (från 0.9.0-beta23). PowerShell avslutas med programmet. Avslutas programmet medan PowerShell skriver, kommer posterna bara till filen (avsnitt 10). Detsamma gäller poster som föll ur kön under en lång störning (avsnitt 7.3). Nästa start som skriver i samma fil skickar de poster som händelseloggen inte kvitterade, med samma rad och hash som förut, och skriver `logg.handelselogg_efterskick` (1007) med antalet. Bara poster som ankaret styrker skickas: programmet följer kedjan bakåt från ankarets post, så en rad som någon annan har lagt in i filen skickas aldrig, så länge ankaret är orört. Ankaret har ingen hemlighet. Den som ändrar både filen och ankaret kan få programmet att skicka en rad som hon själv har skrivit, men det kan hon också göra direkt (avsnitt 7.3). Högst 2 000 poster skickas, de senaste. Ankaret minns de okvitterade posterna som högst 100 intervall av löpnummer per fil. Blir de fler slås de två intervall ihop som ligger närmast varandra, så att ingen okvitterad post glöms. Poster mellan dem, som händelseloggen redan har, skickas då en gång till. Händelseloggen måste vara påslagen vid den starten. Vad som inte skickas står i avsnitt 10. I 0.9.0-beta22 och äldre skickades sådana poster aldrig.

Efterskicket och exporten (från 0.9.0-beta23). Efterskicket går i egna anrop till PowerShell, ett i taget och högst 50 poster per anrop. Ett nytt anrop startar bara när inga andra poster väntar och ingen export väntar på kvitto. En export som börjar medan ett efterskicksanrop pågår väntar inte på det: `export.paborjad` skickas i ett eget anrop samtidigt. De två anropen delar då på datorn, och hur mycket långsammare exportens anrop blir under hård belastning är inte mätt. Som förut kan en export få vänta på ett anrop med andra poster som redan pågår. Exportens väntan är alltså dess eget anrop plus högst ett anrop som redan pågår. Tar varje anrop mer än fem sekunder, som på en hårt belastad dator, räcker tio sekunder inte, och med `RequireEventLogForExport` vägras exporten då. Misslyckas ett efterskicksanrop delas det i mindre. En post som händelseloggen inte tar emot ens ensam, två gånger med ett lyckat anrop emellan, ges upp, och en ny `logg.handelselogg_efterskick` med `ej_skickade` 1 skrivs. Har inget annat anrop lyckats emellan, till exempel för att inga andra poster skrivs, skickar programmet en post som händelseloggen nyss tog emot en gång till, som prov. Lyckas provet ges den vägrade posten upp nästa gång den vägras, och den prövade posten står två gånger i händelseloggen, med samma hash. Misslyckas provet fungerar händelseloggen inte, och inget ges upp. Ett misslyckat efterskick ger ingen `logg.handelselogg_fel`. Efter två misslyckade efterskicksanrop i rad väntar efterskicket 5 sekunder, 30 sekunder, 2 minuter och därefter 5 minuter mellan försöken, och det börjar om när ett anrop lyckas.

Exporten står i händelseloggen innan utfilen skrivs (från 0.9.0-beta22). Sist före motorn, efter programmets egna kontroller, skrivs `export.paborjad` (3006). Den skickas till händelseloggen direkt, och exporten väntar på kvittot från PowerShell, högst tio sekunder. Det gör exporten ungefär en sekund långsammare: i ett prov mot den riktiga händelseloggen fanns posten där 1,1 sekunder efter anropet. Varje påbörjad export följs av `export.lyckad`, `export.stoppad` eller `export.lackkontroll_stopp`. En påbörjad export utan slutpost betyder att programmet avslutades under exporten, till exempel genom att processen dödades, och då kan utfilen finnas.

Kommer inget kvitto inom tio sekunder fortsätter exporten, och posten skrivs till händelseloggen så snart det går. Med policyn `RequireEventLogForExport` = 1 vägras exporten i stället, och `export.stoppad` med orsak `HANDELSELOGG` skrivs i filen. Kravet gäller bara när händelseloggen är påslagen, och inte en start då registret inte gick att läsa (avsnitt 7.1). På en dator där händelseloggen inte fungerar, till exempel för att PowerShell är spärrat för användarna, vägras med policyn alla exporter. Varje export försöker då skriva till händelseloggen igen, och om PowerShell hänger väntar exporten upp till cirka tio sekunder innan den vägras. Pröva därför att händelseloggen fungerar innan policyn sätts.

När loggen startar. Direkt när programmet har startat. Händelser under de första millisekunderna, till exempel licensens första bedömning, hålls i minnet och skrivs efter startposten med sin egen tidpunkt.

3. Format

En rad per post, UTF-8, radslut `\n` (aldrig `\r\n`). Nyckelordningen är fast.

```
{"v":1,"seq":42,"tid":"2026-10-
05T07:41:12.311Z","kod":"export.stoppad","id":3001,"niva":"varning","app":"0.9.0-beta20","dator":"KOMMUN-
PC17","konto":"anna.handlaggare","data":
{"steg":"fore_motor","orsak":"KALLFIL_ANDRAD","dokumenttyp":"pdf","kalla_sha256":"9f2c…"},"prev":"5be1…"}
```

Fält	Innehåll
`v`	Formatversion, 1.
`seq`	Löpnummer i filen, från 1.
`tid`	UTC, ISO 8601 med millisekunder, från datorns klocka.
`kod`	Händelsekod (avsnitt 5).
`id`	Händelse-ID i Windows händelselogg.
`niva`	`info`, `varning` eller `fel`.
`app`	Programversionen som skrev posten.
`dator`	Datornamn.
`konto`	Windows-kontot som körde programmet. Samma uppgift som exportloggens `user_id`.
`data`	Händelsens fält, se avsnitt 5. `avvisade_falt` anger hur många fält som inte släpptes in (avsnitt 6).
`prev`	SHA-256 (hex) av föregående rads exakta UTF-8-bytes, utan radslut. 64 nollor i första posten.

4. Hashkedjan och hur den kontrolleras

Varje post bär hashen av raden före. Radens egen hash står inte i raden; den bärs av nästa post och skickas med till händelseloggen. Det ger:

- En ändrad post, även ett enda blanksteg, bryter länken till nästa rad.
- En borttagen eller omkastad post bryter länken och löpnumret.
- Borttagna poster i början syns på att första raden inte börjar kedjan.
- En fil som sparats om med Windows radslut syns på första raden efter den första.

Kedjan ensam visar inte att de sista posterna har tagits bort, att hela filen har skrivits om med nyräknade hashar eller att en korrekt länkad post har lagts till sist. Därför finns två skydd till.

Ankaret. Vid varje start jämförs filen med sitt ankare, även om loggen emellan har skrivits på en annan plats (reservkatalogen, eller en annan dator med samma roamande profil). Saknas filen, är den kortare än ankaret, eller har posten som ankaret pekar på ändrats, skrivs `logg.kedjebrott` med orsak `fil_saknas`, `fil_kortad` eller `ankarpost_andrad`. Börjar filen inte med exakt de byte programmet har skrivit, rapporteras första brottet i filen, eller `fil_andrad` om kedjan ändå håller. Hela rader efter programmets senaste skrivning som programmet inte skrev ger `frammande_rad`, även när de länkar korrekt. En krasch lämnar högst en ofullständig rad, aldrig en hel. Kedjan fortsätter sedan från filens sista rad. Den som raderar både filen och ankaret syns inte i filen, men mot SIEM. En fil som aldrig har använts på datorn förut prövas bara mot sin egen kedja.

Ankaret ligger i användarens egen datamapp och har ingen hemlighet. Det skyddar mot den som bara kommer åt loggfilen, till exempel med rätten att lägga till data i katalogen i avsnitt 7.2. Användaren själv kan räkna om ankaret. Mot användaren är skyddet att posterna redan finns i SIEM.

Händelseloggen. Samma poster ligger i Windows händelselogg. Där kan användaren inte ta bort enskilda poster, men en vanlig användare kan skriva så många poster till loggen Application att äldre poster skrivs över när loggen når sin maxstorlek, och en administratör kan rensa loggen. Samlar organisationen in händelseloggen till SIEM kan användaren inte ändra det som redan har samlats in. Skyddet beror alltså på att insamlingen sker innan posterna hinner skrivas över.

Ett brott rapporteras en gång, så länge resten av filen är orörd. En rad som avbröts vid ett strömavbrott ger ett `logg.kedjebrott` vid nästa start, inte vid varje start. Ändras något i filen efteråt rapporteras första brottet på nytt. Ett nytt `logg.kedjebrott` med samma rad som ett tidigare betyder alltså att filen har ändrats igen. Den post som står närmast före en trasig rad skyddas inte av länkningen; den finns kvar i händelseloggen.

Verifieringsskript

Itha AB har ett verifieringsskript för Windows PowerShell 5.1 som jämför filen med händelseloggen eller med det SIEM har tagit emot. Det kräver ingen installation och ändrar ingenting. Skriptet lämnas ut på begäran till feedback@privera.se.

```
powershell -NoProfile -ExecutionPolicy Bypass -File .\Verifiera-PriveraSakerhetslogg.ps1 `
  -Path '\\loggserver\privera$\privera-sakerhetslogg-KOMMUN-PC17-anna.handlaggare.jsonl'

# På datorn själv: jämför också med händelseloggen
.\Verifiera-PriveraSakerhetslogg.ps1 -Path <fil> -JamforHandelselogg

# Med senaste hash som SIEM har tagit emot
.\Verifiera-PriveraSakerhetslogg.ps1 -Path <fil> -ForvantadSistaSha256 <hex>

# Med de poster SIEM har tagit emot (en postrad per rad, i mottagningsordning),
# och var programmet skriver filen
.\Verifiera-PriveraSakerhetslogg.ps1 -Path <fil> -MottagnaPoster .\siem-export.jsonl `
  -Loggfil '\\loggserver\privera$\privera-sakerhetslogg-KOMMUN-PC17-anna.handlaggare.jsonl'
```

Vid jämförelsen delas de mottagna posterna i kedjor efter hur de länkar. Samma dator och konto kan ha flera kedjor med samma filnamn, till exempel policyfilen och filen i reservkatalogen. Varje `app.start` bär loggfilens identitet (`loggfil`, SHA-256 av filens verkliga sökväg). Den är densamma för en fil som har raderats och skapats om.

Ange `-Loggfil`. Identiteten i filens egna `app.start` kan användaren skriva själv, eftersom källan `Privera` är öppen för alla (avsnitt 7.3). Med `-Loggfil` anger administratören var programmet skriver filen, som sökvägen ser ut på användarens dator: policyns katalog eller `C:\Users\<konto>\AppData\Roaming\privera-desktop\sakerhetslogg`, plus filnamnet. Ange den verkliga katalogen, inte en junction, ett kort 8.3-namn eller en mappad enhetsbokstav för en lokal katalog. `-Path` får vara en kopia. Utan `-Loggfil` tas identiteten ur filen, och varje kedja med en annan identitet fäller kontrollen.

Skriptet fäller kontrollen när:

- en mottagen post saknas i filen (`HandelserSomSaknasIFilen`),

- en mottagen kedja med filens identitet saknas helt i filen, alltså när filen har raderats och skapats på nytt (`KedjorSomSaknasIFilen`),
- en `app.start` i filen har en annan identitet än `-Loggfil` (`StartposterMedAnnanLoggfil`): filen är omskriven, eller `-Loggfil` är fel angiven,
- en mottagen kedja har en annan identitet och `-Loggfil` saknas (`AndraKedjor`, `AndraKedjorKraverLoggfil`),
- en mottagen kedja utan poster i filen inte går att knyta till någon loggfil (`KedjorUtanLoggfil`); exportera då ett längre tidsfönster,
- en post i filen saknas hos mottagaren fast mottagaren har poster både före och efter den (`PosterSomSaknasHosMottagaren`): posten är inlagd i filen utanför programmet, eller nådde aldrig händelseloggen,
- programmet självt har skrivit `logg.kedjebrott` i filen (`KedjebrottIFilen`), även när kedjan efter brottet håller,
- filen har poster men ingen mottagen kedja motsvarar den (`IngenKedjaMotsvararFilen`): filen kan vara omskriven i sin helhet.

Med `-Loggfil` räknas en kedja med en annan `loggfil`, till exempel reservkatalogens, i `AndraKedjor` utan att fälla kontrollen. Poster efter den sista mottagna redovisas i `PosterEfterSistaMottagna` men fäller inte, eftersom insamlingen kan ligga efter. Exportera alla händelser från källan `Privera` för datorn, inte ett urval av händelse-ID.

Jämförelsen visar borttagna poster och poster som bara finns i filen. Den visar inte att en post som finns både i filen och i händelseloggen skrevs av programmet.

Utgångskod 0 betyder att kedjan håller och jämförelserna stämmer, 1 brott eller avvikelse (första felet anges med radnummer), 2 att filen inte gick att läsa och 3 att PowerShell körs i Constrained Language Mode. Skriptet kräver FullLanguage. Under AppLocker eller WDAC körs det på en administrativ arbetsstation eller tillåts i policyn.

Skriptet är provat mot programmets egen kedja, en ändrad post, en borttagen sista post, två loggplatser med samma filnamn, en fil som har raderats och skapats om (också via en junction), en fil omskriven med påhittad identitet och en korrekt länkad post som lagts in i filen men inte nått SIEM.

Kontroll i SIEM. Varje händelse i händelseloggen bär raden och `sha256=<radens hash>`. Kedjorna följs per dator och konto: varje händelse ska ha en `prev` som är `sha256` för en tidigare mottagen händelse från samma dator och konto, eller 64 nollor och `seq` 1, vilket är början på en ny kedja (en ny loggplats, se `logg.plats_andrad` och `loggkatalog` i `app.start`). En `prev` som inte motsvarar någon mottagen händelse betyder att en post saknas i insamlingen, att filen är raderad eller kortad, eller att en post har lagts in i filen utanför programmet. Två händelser med samma `prev` betyder att filen har kortats och skrivits vidare. Villkoret "nästa händelse har `prev` lika med föregående hash" ger falsklarm när datorn växlar mellan policyns katalog och reservkatalogen.

5. Händelsekoder

Händelse-ID:n är stabila: en kod byter aldrig ID, och nya koder får nya ID.

ID	Kod	Nivå	Fält i `data`	När
1000	`app.start`	info	`utgava` (kund/intern), `paketerad`, `loggkatalog` (standard/policy/reserv), `fillogg`, `loggfil` (SHA-256 av loggfilens verkliga sökväg), `handelselogg`, `rensning_tillaten`, `uppdateringar` (automatisk/kontrollera/av, från 0.9.0-beta21), `ogiltiga_policyvarden`, `policy_lasfel`, `tappade_fore_start`	Varje programstart.
1001	`uppdatering.installerad`	info	`fran`, `till` (versioner)	Första start med en ny version på den här datorn och det här kontot.

ID	Kod	Nivå	Fält i `data`	När
1002	`logg.kedjebrott`	varning	`orsak` (fil_saknas, fil_kortad, ankarpost_andrad, ogiltig_rad, fel_start, fel_lankning, fel_seq, oavslutad_rad, frammande_rad, fil_andrad), `forsta_fel_rad`, `forvantad_seq`, `funnen_seq`	Vid start, när filen inte stämmer med kedjan eller sitt ankare. Rapporteras en gång så länge filen i övrigt är orörd.
1003	`logg.plats_andrad`	info	`till` (standard/policy/reserv)	Första posten när datorn och kontot skriver i en annan fil än förra gången.
1004	`logg.skrivfel`	fel	`mal` (policy), `reserv` (standard)	Policyns katalog gick inte att skriva till.
1005	`logg.handelselogg_fel`	varning	inga	Händelseloggen gick inte att skriva tre gånger i rad. Ett anrop som inte har svarat inom 30 sekunder räknas som misslyckat, så när PowerShell hänger dröjer posten upp till cirka 90 sekunder. Skrivs en gång per störning. Programmet fortsätter att försöka med ökande väntan (avsnitt 7.3), och posten når händelseloggen först när den fungerar igen. Avslutas programmet innan dess skickas posten från 0.9.0-beta23 vid nästa start (avsnitt 2). Före 0.9.0-beta22 slutade programmet försöka för resten av sessionen. Skrivs inte när PowerShell avslutas med programmet mitt i ett anrop.
1006	`app.felsokningsport_tillaten`	varning	inga	Från 0.9.0-beta22. Kundutgåvan startades med en felsökningsport, och policyn `AllowDebugPort` tillät det. Utan policyn avslutas programmet i stället, se IT-drift avsnitt 4.1.
1007	`logg.handelselogg_efterskick`	varning	`antal`, `forsta_seq`, `sista_seq`, `ej_skickade`	Från 0.9.0-beta23. Vid start, när poster i filen inte nådde Windows händelselogg förra gången, till exempel för att programmet avslutades medan PowerShell skrev. Också poster som föll ur kön under en lång störning. `antal` poster skickas igen nu, med samma rad och hash. `forsta_seq` och `sista_seq` är det lägsta och högsta löpnumret bland dem. `ej_skickade` saknades men kunde inte skickas: fler än 2 000, eller raden saknas eller är ändrad i filen. Skrivs också med `antal` 0 och `ej_skickade` 1 när händelseloggen inte tar emot en efterskickad post (avsnitt 2).
2000	`licens.aktiverad`	info	`overtagning`	Licensen aktiverades på datorn.
2001	`licens.aktivering_misslyckad`	varning	`felkod`	Felskriven nyckel, okänd nyckel, platstak, ingen kontakt med mera. `wind_down`: försök att aktivera i avvecklingsläget.
2002	`licens.avaktiverad`	info	inga	Datorn kopplades loss.
2003	`licens.status_andrad`	info	`fran`, `till` (licenstillstånd), `far_kora`	Första bedömningen vid start (`fran` = start) och varje ändring: utgången, klocka bakåt, fel dator, förnyad.

ID	Kod	Nivå	Fält i `data`	När
2004	`licens.fornyelse_misslyckad`	varning	`felkod`	Förnyelse nekad av servern. Utebliven kontakt loggas inte här; den syns som tillståndsändring när den får betydelse.
2005	`licens.avaktivering_misslyckad`	varning	`felkod`	Frånkoppling utan kontakt med servern, eller (`wind_down`) i avvecklingsläget.
2006	`avveckling.besked`	varning	inga	Ett signerat avvecklingsbesked gäller: vid första bedömningen varje start i avvecklingsläget och när ett besked börjar gälla. Se Avvecklingsläge.
2007	`avveckling.besked_avvisat`	varning	`kalla` (fil/server)	Ett avvecklingsbesked gick inte att verifiera och avvisades. Filens innehåll loggas inte.
2008	`licens.ej_sparad`	varning	`klartextfil_kvar`	Licensen kunde inte sparas krypterad och finns bara i minnet. Nyckeln skrivs aldrig i klartext. `klartextfil_kvar` är sant när en licensfil i klartext från en äldre version ligger kvar på disken.
3000	`export.lyckad`	info	`dokumenttyp` , `maskeringar` , `omaskerbara` , `borttagna_bilder` , `stil` (box/text), `lackkontroll` (godkand), `exportlogg` (app/bredvid), `kalla_sha256` , `utfil_sha256`	Export klar.
3001	`export.stoppad`	varning	`steg` (fore_spardialog, fore_motor, motor), `orsak` , `dokumenttyp` , `kalla_sha256`	Export stoppad, se orsaker nedan.
3002	`export.lackkontroll_stopp`	varning	`steg` (motor), `orsak` (LACKKONTROLL), `dokumenttyp` , `kalla_sha256`	Motorns efterkontroll hittade ett maskerat värde kvar i utfilen. Ingen fil skrevs.
3003	`historik.rensad`	varning	`antal` , `mapp` (app/vald)	Användaren rensade exportloggar med "Rensa mappen".
3004	`historik.rensning_nekad`	varning	inga	"Rensa mappen" nekad av policyn `AllowClearExportLogs=0` .
3005	`granskningar.rensade`	info	`antal`	Sparade granskningar rensades i Inställningar.
3006	`export.paborjad`	info	`dokumenttyp` , `dokument` (antal källfiler), `kalla_sha256`	Från 0.9.0-beta22. Sist före motorn, innan utfilen skrivs. Exporten väntar på kvitto från händelseloggen (avsnitt 2).
3007	`analys.klar`	info	`dokumenttyp` , `sidor` , `fynd` (antal), `kalla_sha256`	Från 0.9.0-beta22. Motorn har analyserat ett dokument. Fyndens text loggas inte.
3008	`analys.nekad`	varning	`orsak` (LICENS, SOKVAG), `dokumenttyp`	Från 0.9.0-beta22. En analys nekades innan motorn startade: licensen tillåter inte körning, eller sökvägen godtas inte. Sökvägen loggas inte.
4000	`installning.andrad`	info	`nyckel` (exportlogg_bredvid_utfil, ner, tvingad_ocr, forvalda_kategorier), `varde`	En säkerhetsrelevant inställning ändrades.
5000	`uppdatering.hamtad`	info	`version`	En uppdatering är hämtad och redo.
5001	`uppdatering.installation_startad`	info	inga	Användaren valde att installera nu. Att installationen gick igenom syns som 1001 vid nästa start.

ID	Kod	Nivå	Fält i `data`	När
5002	`uppdatering.fel`	varning	inga	Uppdateringskontrollen misslyckades. Felets text loggas inte.
5003	`uppdatering.signatur_nekad`	varning	`orsak` (ogiltig, fel_utgivare, fel_fil, kontrollen_kunde_inte_koras)	Från 0.9.0-beta22. En hämtad uppdatering avvisades: signaturen är inte giltig, utgivaren är en annan, signaturen eller versionsuppgiften gäller inte den hämtade filen eller saknas, eller kontrollen kunde inte köras. Skrivs också vid start när kontrollen inte gick att sätta upp, och då görs ingen uppdateringskontroll. Se IT-drift avsnitt 5.

Orsaker i `export.stoppad`

Orsak	Steg	Betydelse
`LICENS`	fore_motor	Licensen tillåter inte körning.
`SOKVAG`	fore_spardialog, fore_motor	En sökväg underkändes (relativ, nätverk, fel filtyp).
`FINGERAVTRYCK_SAKNAS`	fore_spardialog, fore_motor	Analysen saknar källfilens SHA-256.
`KALLFIL_ANDRAD`	fore_spardialog, fore_motor	Källfilen har ändrats sedan analysen.
`KALLFIL_OLASBAR`	fore_spardialog, fore_motor	Källfilen gick inte att läsa (flyttad, raderad, låst).
`UTFIL_AR_KALLFIL`	fore_motor	Användaren valde källfilen som utfil.
`MOTORFEL`	motor	Motorn svarade inte med en rapport (krasch, avbrott).
`GEOMETRI`, `ETIKETT`, `TOLKNING`, `KALLFIL_ANDRAD`, `OVRIGT`	motor	Motorns egna stopp. `OVRIGT` omfattar bland annat Word-dokumentets paketinventering och fel utan egen felklass.
`HANDELSELOGG`	fore_motor	Från 0.9.0-beta22. Policyn `RequireEventLogForExport` kräver kvitto från händelseloggen, och det kom inget (avsnitt 2).
`OKAND`	motor	Motorn stoppade utan kod.

`LACKKONTROLL` har en egen händelse (3002), så att SIEM kan larma på just den.

6. Vad som aldrig loggas

Varje händelse har en fast lista med typade fält: sant eller falskt, antal, SHA-256, versionsnummer eller ett värde ur en fast lista. Det finns inga fritextfält. Ett fält som inte står i listan, eller ett värde av fel typ, kommer inte in i posten; det räknas i `avvisade_falt`. Därför står aldrig följande i säkerhetsloggen:

- dokumentinnehåll, fyndens text, etiketter eller sidpositioner,
- filnamn och sökvägar, inte heller hashade, eftersom ett filnamn går att gissa fram ur sin hash. Enda undantaget är säkerhetsloggens egen fil: `loggfil` i `app.start` är SHA-256 av dess sökväg, som består av loggkatalogen samt datornamn och konto, som redan står i posten,
- felmeddelanden, eftersom de kan bära filnamn och sidnummer,
- licensnyckel, e-post, licens-id eller maskin-id.

Dokument identifieras med SHA-256 av filens innehåll, samma värde som i exportloggen. Innehållet går inte att räkna fram ur hashen, men den som har en fil kan visa om det var just den som exporterades.

Det som är personuppgift i posten är Windows-kontots namn och datornamnet. De finns med för att loggen ska kunna svara på vem som gjorde vad.

Automatiska tester planterar personnummer, namn, sökvägar och felmeddelanden i varje exportväg och kontrollerar att inget av det hamnar i fil eller händelselogg.

7. Organisationens inställningar

7.1 Policy i registret

Nyckeln `HKLM\Software\Policies\Privera` läses vid start med `reg.exe` (64-bitarsvyn). Nyckeln är bara skrivbar för administratörer.

Från 0.9.0-beta22 startar programmet `reg.exe`, och Windows PowerShell för händelseloggen och för signaturkontrollen av uppdateringar, med full sökväg i katalogen `System32` under Windows-katalogen. Katalogen tar programmet ur den plats som Windows laddade systembiblioteket `ntdll.dll` från, inte ur miljövariabeln `SystemRoot`, och programmen startas med en egen, minimal miljö i stället för processens. Variabler som `SystemRoot` och `PSModulePath` styr alltså inte längre vilket program som svarar eller vilken kod PowerShell laddar.

Före 0.9.0-beta22 togs katalogen ur `SystemRoot`, som programmet ärver från den som startar det. En användare som kunde köra egna program kunde därför låta ett annat program svara i stället för `reg.exe` eller PowerShell, och på så sätt få programmet att använda andra policyvärden eller att inte skriva till händelseloggen. I de versionerna kan den signerade programfilen dessutom användas för att köra egen kod (se IT-drift avsnitt 4.1).

Registerpolicyn är ändå en styrning för vanlig användning, inte ett skydd mot en användare som avsiktligt vill kringgå den. I en installation per användare kan användaren byta ut programmets filer (se IT-drift avsnitt 4.1), och en katalog av typen `REG_EXPAND_SZ` i `SecurityLogDirectory` expanderas med användarens egna miljövariabler.

Värde	Typ	Förval	Betydelse
`SecurityLogEnabled`	REG_DWORD	1	0 = ingen loggfil. Händelseloggen skrivs ändå om den är på.
`SecurityLogDirectory`	REG_SZ eller REG_EXPAND_SZ	programmets datamapp	Katalog för loggfilen. Absolut lokal sökväg eller UNC, utan `..`.
`EventLogEnabled`	REG_DWORD	1	0 = inget till Windows händelselogg.
`AllowClearExportLogs`	REG_DWORD	1	0 = "Rensa mappen" i Historik nekas och loggas.
`RequireEventLogForExport`	REG_DWORD	0	Från 0.9.0-beta22. 1 = en export vägras om `export.paborjad` inte fick kvitto från händelseloggen (avsnitt 2). Gäller bara när `EventLogEnabled` är 1.
`UpdateMode`	REG_DWORD	2	Från 0.9.0-beta21. 0 = av, 1 = kontrollera och meddela, 2 = automatisk. Ett ogiltigt värde ger av. Se IT-drift avsnitt 5.

Ett värde av fel typ ignoreras och förvalet gäller (utom för `UpdateMode`, se tabellen); antalet står i `app.start` som `ogiltiga_policyvarden`. Saknas nyckeln gäller förvalen. Gick registret inte att läsa (tidsgräns på fem sekunder, eller `reg.exe` saknas eller startar inte) gäller förvalen för loggningen, aldrig avstängd logg, men "Rensa mappen" nekas och `app.start` får `policy_lasfel` = true. Förvalen gäller då också `RequireEventLogForExport`, så kravet på kvitto gäller inte den starten. "Rensa mappen" väntar också in policyn: har den inte lästs inom tio sekunder från klicket nekas rensningen.

```
reg add HKLM\Software\Policies\Privera /v SecurityLogDirectory /t REG_SZ /d "\\loggserver\privera$" /f
reg add HKLM\Software\Policies\Privera /v AllowClearExportLogs /t REG_DWORD /d 0 /f
```

Samma värden kan sättas med grupprincip eller Intune, med ADMX-mallen som beskrivs i IT-drift avsnitt 6 eller som registerinställning. Policyn läses vid start, och en ändring gäller från nästa start.

`AllowClearExportLogs=0` stoppar bara programmets egen rensning. Filer kan fortfarande tas bort i Utforskaren om användaren har behörighet till mappen.

7.2 Append-only-katalog för loggfilen

Programmet öppnar loggfilen bara för tillägg. Då räcker rätten att lägga till data (Append Data), och organisationen kan neka användaren rätten att skriva om, radera eller byta namn på filen. Katalogen ska skapas av en administratör, så att användaren inte äger den.

```
mkdir C:\ProgramData\Privera\Sakerhetslogg
icacls C:\ProgramData\Privera\Sakerhetslogg /inheritance:r ^
  /grant:r "SYSTEM:(OI)(CI)F" "BUILTIN\Administratörer:(OI)(CI)F" ^
  /grant:r "BUILTIN\Användare:(RX,WD,AD)" ^
  /grant:r "BUILTIN\Användare:(OI)(IO)(RX,AD,WA,WEA)" ^
  /grant:r "*S-1-3-4:(OI)(IO)(RX,AD,WA,WEA)"
```

`S-1-3-4` (Ägarrättigheter) tar bort den rätt att ändra behörigheten som filens ägare annars har. På en svensk Windows heter grupperna `BUILTIN\Administratörer` och `BUILTIN\Användare`. SID:erna `*S-1-5-32-544` och `*S-1-5-32-545` fungerar på alla språk.

Provat på ett administratörskonto utan förhöjning (UAC-filtrerad token), inte på ett rent standardkonto: programmets sätt att skriva skapade filen och lade till rader, medan överskrivning, radering och namnbyte nekades, och kontot kunde inte ändra filens behörighet. Resultatet bör gälla ett standardkonto likadant, men det är inte provat. Att användaren inte heller kan ändra katalogens behörighet förutsätter att katalogen ägs av en administratör. Det är inte provat.

7.3 Registrering av händelsekällan

Registreringen är valfri och kräver administratör en gång. Programmet skriver till händelseloggen utan administratörsrätt och utan registrering. Oregistrerat visar Loggboken raden "Beskrivningen för händelse-ID … hittades inte", men källa, ID, nivå och posten (i EventData) är korrekta, och det är det SIEM läser.

Registreras källan vid utrullningen visas posten som meddelandetext:

```
New-EventLog -LogName Application -Source Privera
```

Constrained Language Mode (AppLocker, WDAC). Utan registrerad källa skriver programmet via Add-Type, som är spärrat där. Efter tre misslyckade försök i rad skrivs `logg.handelselogg_fel`, och programmet väntar sedan 5 sekunder, 30 sekunder, 2 minuter och därefter 5 minuter mellan nya försök. En export gör alltid ett nytt försök direkt. Under en lång störning hålls högst 2 000 poster i kö till händelseloggen; äldre poster faller då bort ur kön till händelseloggen men står kvar i filen, om filloggen är på. Från 0.9.0-beta23 skickas de av nästa start som skriver i samma fil, högst 2 000 (avsnitt 2). Före 0.9.0-beta22 slutade programmet försöka för resten av sessionen. I låsta miljöer behöver källan registreras. Med registrerad källa anropar programmets PowerShell-skript inga .NET-metoder före `Write-EventLog`. Det är provat i en PowerShell-session i Constrained Language Mode med `Write-EventLog` utbytt mot en provfunktion: id, nivå och text kom fram tecken för tecken, även å, ä, ö, radbrytningar och citattecken. Det är inte provat under en riktig AppLocker- eller WDAC-policy med registrerad källa, och inte heller att en standardanvändare kan skriva till en registrerad källa.

Vem som kan skriva med källan `Privera`. Att programmet kan skriva utan administratörsrätt betyder att varje användare kan det, med eller utan registrerad källa. En händelse med källan `Privera` bevisar alltså inte att programmet skrev den. Det som går att lita på är att det som organisationens SIEM har samlat in inte kan tas bort av användaren. I den lokala händelseloggen kan användaren inte ta bort enskilda poster, men kan få äldre poster att skrivas över, och en administratör kan rensa loggen. Ingen av dem visar att en post är äkta (avsnitt 4).

Installern registrerar inte källan, eftersom den förvalt körs utan administratörsrätt. Lägg raden i organisationens utrullningsskript (Intune, Configuration Manager eller grupprincip). Ett exempel står i IT-drift avsnitt 2.5.

8. Så läser SIEM loggen

Exemplen i avsnittet är inte provade. Insamlingen är inte provad med Windows Event Forwarding, Microsoft Sentinel, Splunk eller något annat SIEM.

Windows händelselogg (rekommenderat). Samla in loggen *Application* med filtret källa `Privera`. Med Windows Event Forwarding:

```
<QueryList><Query Id="0" Path="Application">
  <Select Path="Application">*[System[Provider[@Name='Privera']]]</Select>
</Query></QueryList>
```

Microsoft Sentinel (Azure Monitor Agent, datainsamlingsregel med samma XPath) lägger händelserna i tabellen `Event`:

```
Event
| where Source == "Privera"
| extend Rad = tostring(split(RenderedDescription, "\nsha256=")[0])
| extend Post = parse_json(extract(@"(\{.*\})", 1, Rad))
| project TimeGenerated, Computer, EventID, Kod = tostring(Post.kod), Seq = toint(Post.seq), Konto =
tostring(Post.konto), Data = Post.data
```

Hur `RenderedDescription` och `EventData` ser ut beror på om källan är registrerad och på agenten. Kontrollera mot en första insamlad händelse. Med oregistrerad källa är nivåns namn tomt, eftersom ingen meddelandefil finns. Filtrera därför på nivåns tal (`Level`: 2 = fel, 3 = varning, 4 = information) eller på händelse-ID, inte på nivåns namn.

Splunk Universal Forwarder: `[WinEventLog://Application]` med `whitelist = SourceName=Privera`.

Händelse-ID att larma på: 1002, 1004, 1005, 1006, 3002, 3004 och 5003, samt 2003 med `till` = `clock_tampered` eller `machine_mismatch` och 3001 med `orsak` = `HANDELSELOGG`. En 3006 som inte följs av 3000, 3001 eller 3002 från samma dator och konto betyder att programmet avslutades under en export.

Filen. Peka `SecurityLogDirectory` mot en katalog som SIEM-agenten läser (Splunk `[monitor://…\*.jsonl]`, NXLog `im_file`, Filebeat eller Elastic Agent `filestream` med JSON-avkodning). Filerna växer bara genom tillägg och roteras inte.

Programmet skickar ingenting över nätet för loggningen. Insamlingen sker med organisationens egna verktyg.

9. Exportloggen (`.audit.json`)

Exportloggen per export finns kvar: källa och utfil, SHA-256 för båda, tider, Windows-konto, varje maskering med kategori och position, och fel. Den är lika känslig som källdokumentet och ska inte till SIEM.

Nytt från 0.9.0-beta20 är fältet `stopp_kod` när en export stoppas i motorn: `LACKKONTROLL`, `GEOMETRI`, `KALLFIL_ANDRAD`, `ETIKETT`, `TOLKNING` eller `OVRIGT`. Koden sätts ur felets klass, aldrig ur felets text. Fältet är additivt, och `schema_version` är fortfarande 5. Säkerhetsloggen hänvisar till exportloggen genom källfilens SHA-256.

10. Begränsningar

- **Versioner.** Säkerhetsloggen finns från 0.9.0-beta20. Äldre versioner har bara exportloggen och loggar inte stoppade exportförsök eller andra säkerhetshändelser. Händelserna 1006, 3006, 3007, 3008 och 5003, kvittot före export och policyn `RequireEventLogForExport` finns från 0.9.0-beta22.
- **Kedjan har ingen hemlighet.** En användare med skrivrätt till både loggfilen och programmets datamapp kan skriva om allt lokalt. Skyddet mot borttagning är append-only-katalogen (7.2) och händelseloggen i SIEM, inte filen. Mot inlagda poster finns inget fullständigt skydd: en användare kan skriva en korrekt länkad post både i filen och i händelseloggen, och den går inte att skilja från programmets. Programmet upptäcker rader i filen som det inte skrev så länge ankaret är orört, och verifieringsskriptet fäller poster som bara finns i filen när SIEM har poster efter dem.

- **Filens identitet är en sökväg.** Junctions, symboliska länkar, `subst` och korta namn löses upp lokalt, men inte nätverkssökvägar: samma resurs under två UNC-namn, eller via en mappad enhet och UNC, ger två identiteter. Användaren kan också peka om `%APPDATA%` till en annan katalog. Verifieringsskriptet fäller båda fallen när `-Loggfil` anger var programmet ska skriva.
- **Händelseloggen utan registrerad källa kräver PowerShell med Add-Type.** I Constrained Language Mode misslyckas den vägen (7.3). Den oregistrerade vägen är provad på riktigt; den registrerade bara i en provsession.
- **Verifieringsskriptet kräver FullLanguage.** I Constrained Language Mode avslutas det med kod 3.
- **Två exportstopp rapporteras av programmets gränssnitt** (saknat fingeravtryck före spardialogen och utfil som är källfilen). Kod som har tagit över gränssnittet kan låta bli att rapportera dem. Övriga stopp avgör programmets huvudprocess själv.
- **Inställningsändringar rapporteras av gränssnittet**, med samma förbehåll. Exportloggens plats, som är den inställning som påverkar spårbarheten, står dessutom i varje `export.lyckad`, satt av huvudprocessen.
- **Avslut.** Vid ett ordnat avslut väntar programmet högst tre sekunder på att det som har registrerats strax före avslutet ska nå händelseloggen. Från 0.9.0-beta23 väntar avslutet också in en start av loggen som pågår. PowerShell avslutas med programmet. Hinner anropet inte bli klart, eller avslutas programmet utan ordnat avslut (Windows stängs av, användaren loggar ut, strömavbrott, dödad process), finns de sista posterna bara i filen, och ingen `logg.handelselogg_fel` skrivs. SIEM ser då en `prev` som saknas. I prov i Windows Sandbox under hård belastning saknades efter tätt upprepade avslut upp till 19 av 64 poster i händelseloggen, fast kedjan i filen höll (2026-10-29). I 0.9.0-beta22 och äldre kommer de posterna aldrig fram. Från 0.9.0-beta23 skickas de av nästa start som skriver i samma fil, och `logg.handelselogg_efterskick` (1007) säger hur många (avsnitt 2). Luckan finns kvar tills dess. En post som PowerShell hann skriva men inte kvittera står då två gånger i händelseloggen, med samma hash. Inget skickas igen om ankaret är raderat, om händelseloggen är avstängd vid nästa start eller om programmet inte startas igen. En start i reservkatalogen skickar inte policykatalogens poster. Det gör nästa start som skriver i policykatalogen. Där kedjan i filen är bruten slutar programmet att följa den bakåt. Efter en misslyckad skrivning till filen skickas därför inte de okvitterade poster som står före luckan, och inte heller poster före en ändrad, borttagen eller omflyttad rad. De räknas i `ej_skickade`. Har datorn kört 0.9.0-beta22 emellan är uppgiften om vilka poster som är okvitterade borta, eftersom den versionen skriver ankaret utan den. Poster som 0.9.0-beta22 eller äldre skrev skickas inte heller. Undantaget är `export.paborjad`, som från 0.9.0-beta22 når händelseloggen innan utfilen skrivs, om kvittot kommer inom tio sekunder (avsnitt 2).
- **Kvittot stoppar inte exporten förvalt.** Utan `RequireEventLogForExport` = 1 fortsätter en export som inte fick kvitto inom tio sekunder. Kravet gäller inte heller en start då registret inte gick att läsa (avsnitt 7.1).
- **Tiden kommer från datorns klocka.** Händelseloggens egen tidsstämpel och SIEM:s mottagningstid är oberoende av den.
- **Händelseloggen kan fyllas och rensas.** En vanlig användare kan skriva till loggen Application och få äldre poster att skrivas över, och en administratör kan rensa den. Det som redan har samlats in till SIEM påverkas inte.
- **Policyn går att kringgå för den som kan ändra programmets filer.** Från 0.9.0-beta22 väljer miljövariablerna inte längre vilket `reg.exe` eller PowerShell som startas. I en installation per användare kan användaren ändå byta ut programmets filer, och `REG_EXPAND_SZ` expanderas med användarens miljö (avsnitt 7.1). I 0.9.0-beta21 och äldre startades `reg.exe` och PowerShell från den Windows-katalog som miljövariabeln `SystemRoot` anger.
- **Inte provad mot SIEM.** Insamlingen är inte provad med Windows Event Forwarding eller i något SIEM.
- **Licensserverns säkerhetslogg sparas inte.** Raderna syns bara medan de skrivs och når inte organisationens SIEM (avsnitt 11).
- Ingen rotation och ingen gallring av loggfilen, och ingen logg av att programmet avslutas. Analys av dokument loggas från 0.9.0-beta22 (3007 och 3008).

11. Licensservern

Licensservern licens.privera.se drivs av Itha AB hos Cloudflare (Workers) och sköter aktivering, förnyelse, uppdateringar och nedladdning av installern. Den tar inte emot dokument. Säkerhetsloggen ovan gäller programmet på datorn. Det här avsnittet gäller licensservern.

Vad som loggas. Licensservern skriver en rad i JSON per säkerhetshändelse: aktivering, flytt av licensen till en annan dator, förnyelse, frånkoppling, ändrad licensstatus efter avstämningen mot Stripe, hämtning av licensnyckeln efter köpet, inloggning på kundsidan med licensnyckeln, nedladdning av installer och uppdateringar, anrop från Stripes webhook (med utfallet nekad när signaturen inte stämmer), administrativa anrop, taktbegränsning, oväntade serverfel och anrop till sökvägar som inte finns. Varje rad har händelsekod, nivå (info, varning eller fel), utfall (lyckad, nekad eller fel), orsak, HTTP-status och Cloudflares id för anropet, och när de är kända licensens id i licensdatabasen och Stripes händelse-id.

Vad som aldrig loggas. Licensnyckeln och klientens IP-adress står bara som pseudonym: HMAC-SHA-256, avkortad till 64 bitar, med en nyckel som bara finns i licensservern. Samma nyckel eller adress ger samma pseudonym, så upprepade försök syns, men pseudonymen går inte att vända utan nyckeln. E-postadresser, kassans session-id, felmeddelanden, anropens innehåll och sökvägen i ett anrop till en okänd adress loggas inte. Värdena prövas mot fasta mönster, och ett värde som ser ut att bära en licensnyckel kommer inte med.

Driftloggen. Felmeddelanden i licensserverns driftlogg tvättas innan de skrivs: licensnyckeln i alla former som servern godtar, e-postadresser och IP-adresser byts mot en markering, och av kassans session-id står bara de sista åtta tecknen kvar.

Raderna sparas inte. Cloudflares lagring av loggar (Workers Logs) är avstängd för licensservern. Skälet är att licensnyckeln och kassans session-id fortfarande skickas i adressen vid nedladdningen av installern (`/v1/installer?key=` och `?id=`) och vid hämtningen av nyckeln efter köpet (`/v1/session?id=`), och Cloudflare kan spara adressen tillsammans med loggraderna. Säkerhetsraderna och driftloggen syns därför bara medan de skrivs, för Itha AB i Cloudflares realtidsvisning. Ingenting av det sparas, och ingenting når organisationens SIEM.

Planerat. Att flytta licensnyckeln och session-id:t ur adressen och därefter slå på lagringen. Export till organisationens SIEM kräver dessutom Cloudflares betalda Workers-abonnemang och Logpush. Inget datum är fastställt.

Det som finns i licensdatabasen. Licensdatabasen visar vilka licenser som har hämtat installationsfiler och vilka datorer som är aktiverade. Det är inte en logg över misslyckade försök.

Kryptering och åtkomstkontroll

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/kryptering-och-atkomst/

Beskriver kryptering i transit och i vila, nyckelhantering och hur åtkomsten till Privera styrs, och varför programmet saknar egen inloggning och egna roller.

1. Sammanfattning

Krav	Läge
TLS 1.2 som lägsta nivå i transit	Uppfyllt för licens.privera.se och privera.se. TLS 1.0 och 1.1 nekas (kontrollerat 2026-10-23).
TLS 1.3	Stöds av licensservern och webbplatsen.
HSTS	privera.se skickar HSTS med `includeSubDomains`, men finns inte på webbläsarnas förinstallerade lista. licens.privera.se skickar `Strict-Transport-Security: max-age=31536000; includeSubDomains` på varje svar, se avsnitt 2.
Kryptering i vila	Delvis. Hos Cloudflare AES-256. På datorn krypteras sparade granskningar och licensfilen med AES-256-GCM under en nyckel som skyddas av Windows DPAPI. Exporter, exportloggar, säkerhetsloggen, tillfälliga arbetsfiler i `%TEMP%` och senast använda mappar är okrypterade (avsnitt 3).
Krypteringsnivå efter informationsklass (KLASSA)	Saknas. Programmet har ingen inställning för krypteringsnivå. Skyddet av filerna styrs av var organisationen lagrar dem.
Flerfaktorsautentisering (MFA)	Saknas i programmet, som inte har någon egen inloggning. Åtkomsten följer Windows-kontot och därmed organisationens egen inloggning och MFA. Webbens kundsida öppnas med licensnyckeln som enda faktor (avsnitt 5).
Rollbaserad behörighet (RBAC)	Saknas i programmet, som inte har några roller. Vem som får installera och köra programmet styr organisationen med sina AD-grupper, AppLocker, Intune eller grupprincip (avsnitt 5).

2. Kryptering i transit

Dokument skickas aldrig över nätet. Den enda trafiken är licens, uppdateringar och en engångshämtning av språkmodellen. Från 0.9.0-beta22 kontaktar motorn inte Hugging Face när språkmodellen redan finns på datorn. I 0.9.0-beta14 till 0.9.0-beta21 kunde motorn ändå hämta en lista från `huggingface.co/api/agent-harnesses`, högst en gång per dygn, även när modellen fanns på datorn.

Förbindelse	Protokoll	Kommentar
Programmet till licens.privera.se (licens och uppdateringar)	HTTPS, TLS 1.2 eller 1.3	Adressen är inbyggd med `https://` och kan inte pekas om i kundversionen. Kontrollerat 2026-10-23: TLS 1.0 och 1.1 nekas, TLS 1.2 (ECDHE-ECDSA-CHACHA20-POLY1305) och TLS 1.3 (TLS_AES_256_GCM_SHA384) accepteras, och `http://` svarar med omdirigering till `https://`. Varje svar från licensservern har `Strict-Transport-Security: max-age=31536000; includeSubDomains` (utan `preload`), `X-Content-Type-Options: nosniff`, `Referrer-Policy: no-referrer` och `Content-Security-Policy: default-src 'none'; frame-ancestors 'none'; base-uri 'none'; form-action 'none'`. Programmet använder bara `https://`.
Programmet till Hugging Face (språkmodellen, första gången den behövs)	HTTPS	TLS-konfigurationen ägs av Hugging Face, inte av Itha AB. Modellen är låst till en bestämd revision.
Webbplatsen privera.se	HTTPS med HSTS	`max-age=63072000; includeSubDomains; preload`. Domänen finns inte på webbläsarnas förinstallerade HSTS-lista, trots `preload` i huvudet. TLS 1.0 och 1.1 nekas (kontrollerat 2026-10-23).
Licensservern till Stripe	HTTPS	api.stripe.com
Programfönstret till motorn på datorn	Lokala rör (stdio)	Ingen nätverksport öppnas.

3. Kryptering i vila

3.1 På organisationens datorer

Data	Skydd	Kommentar
Originalfiler och exporterade filer	Inget från programmet	Ligger där användaren väljer. Okrypterade med avsikt, så att andra program kan öppna dem.
Exportloggar (`*.audit.json`)	Inget från programmet	Innehåller inte fyndens originaltext, men filnamn, sökvägar, Windows-kontot, ersättningstexter och var varje maskering sitter (sida, teckenintervall och rutor). Positionerna visar var i dokumentet något har maskerats.
Säkerhetslogg (`privera-sakerhetslogg-*.jsonl`)	Inte krypterad. En SHA-256-hashkedja upptäcker ändringar så länge ankaret är orört, men hindrar dem inte. Katalogen kan styras med policy och göras append-only.	Inga dokumentuppgifter, filnamn eller sökvägar, men Windows-kontots namn, datornamn och innehållshashar. Se Säkerhetsloggning.
Tillfälliga arbetsfiler (`%TEMP%\privera_ocr_*`)	Inte krypterade	Motorn lägger sina tillfälliga filer här: sidbilder för textigenkänningen, omaskerade kopior av PDF-filen under bearbetningen och exportens arbetsfil innan den flyttas till sin plats. Filerna visar dokumentets innehåll. De raderas när programmet avslutas, och kataloger som blivit kvar efter en krasch rensas vid nästa start. Raderingen är en vanlig radering, inte en säker överskrivning.
Senast använda mappar	Inte krypterade	Mappen i filutforskaren, mappen för utfiler och mappen för exportloggarnas historik sparas som sökvägar i programfönstrets lokala lagring i Windows-kontots profil.
Sparade granskningar	Electron safeStorage: AES-256-GCM med en nyckel som skyddas av Windows DPAPI och är bunden till Windows-kontot	Programmet sparar inte om skyddet saknas.
Licensfil (licensnyckel och licensbevis)	Electron safeStorage: AES-256-GCM med en nyckel som skyddas av DPAPI	Från 0.9.0-beta20 skrivs nyckel och licensbevis bara krypterade. Saknas krypteringsskyddet hålls licensen i minnet i stället (avsnitt 6).

För högre informationsklasser enligt KLASSA rekommenderar vi att mappar med original, exporter, exportloggar och säkerhetslogg ligger på en krypterad volym, till exempel BitLocker med AES-128 eller AES-256 enligt organisationens inställning, och att systemdisken, där `%TEMP%` och profilen ligger, är krypterad. Programmet har ingen egen inställning för krypteringsnivå.

3.2 Hos Itha AB:s underleverantörer

Data	Plats	Skydd
Licensdatabas (Cloudflare D1)	Region Östeuropa (EEUR). Ingen EU-jurisdiktion är satt, se Underleverantörer.	AES-256 i vila enligt Cloudflare. Licensnyckeln lagras som SHA-256-summa och fyra sista tecken. Under hämtningsfönstret efter köpet finns den dessutom krypterad med AES-256-GCM (avsnitt 6). IP-adresser lagras som HMAC-SHA-256-pseudonym.
Installationsfiler (Cloudflare R2)	Region Östeuropa (EEUR)	AES-256-GCM i vila enligt Cloudflare. Inga kunduppgifter.
Licensserverns loggar (Cloudflare Workers Logs)	Sparas inte	Cloudflares lagring av loggar är avstängd för licensservern, eftersom licensnyckeln och kassans session-id fortfarande står i adressen vid två anrop. Se Säkerhetsloggning avsnitt 11.
Betaluppgifter (Stripe)	Se Underleverantörer	Stripes eget ansvar. Kortuppgifter når aldrig Itha AB.

4. Nyckelhantering

Nyckel	Syfte	Förvaring	Byte
Licensserverns signeringsnyckel (Ed25519, privat)	Signerar licensbevis	Hemlighet i licensservern hos Cloudflare, aldrig i källkoden	Den publika halvan är inbyggd i programmet, så ett byte kräver en ny programversion. Skriftlig rutin saknas (Planerat).
Avvecklingsnyckeln (Ed25519, privat)	Signerar det avvecklingsbesked som låter programmet köra utan licensserver om Itha AB upphör	Skild från licensservern och dess nycklar. Ska flyttas till förvaring offline; per 2026-10-23 är det inte gjort, och nyckeln ligger på en av Itha AB:s datorer.	Byte kräver en ny programversion. Den gamla nyckeln sparas så länge versioner med den används.
Kodsigneringscertifikat (Itha AB)	Signerar installer och programfiler	Microsoft Azure Trusted Signing. Tjänsten lämnar aldrig ut certifikatets privata nyckel; Microsoft anger FIPS 140-3 nivå 3.	Certifikaten förnyas av tjänsten.
Övriga hemligheter i licensservern	Betalning (Stripe), administrativa funktioner, pseudonymisering av IP-adresser och, i säkerhetsloggen, av licensnycklar, kryptering av licensnyckeln efter köp och driftlarm	Hemligheter i licensservern hos Cloudflare	Manuellt.
Kundens licensnyckel	Behörighet för aktivering, förnyelse, uppdatering och nedladdning medan licensen gäller, och för webbens kundsida så länge licensen finns	Hos kunden (DPAPI). På servern som SHA-256-summa.	Kan bytas per kund.

5. Åtkomst: Windows-kontot i stället för egen inloggning och egna roller

Privera är ett lokalt program för en användare åt gången, utan egen inloggning i programmet och utan server som tar emot dokument.

- **Åtkomsten följer Windows-kontot.** Den som är inloggad i Windows kan använda programmet. Programmet lägger inget eget lager ovanpå Windows-inloggningen, så organisationens MFA, lösenordsregler och skärmlås är det som skyddar även programmet. Sparade granskningar och licensfilen är bundna till Windows-kontot via DPAPI. DPAPI skyddar inte mot kod som körs i användarens session, och i en domän kan nycklarna återställas med domänens reservnyckel. Exporter och exportloggar är okrypterade och kan läsas av den som har åtkomst till filerna (avsnitt 3.1).
- **Det finns inga roller att tilldela.** Alla användare har samma funktioner: öppna, granska, maskera och exportera ett dokument på sin egen dator. Vem som får installera och köra programmet styr organisationen med sina egna verktyg, till exempel AD-grupper, AppLocker, Intune och grupprincip. Säkerhetsloggning och uppdateringar styrs med registerpolicy och ADMX-mall, se IT-drift. Exportloggen anger vilket Windows-konto som gjorde exporten, och säkerhetsloggen anger konto och dator för varje händelse.
- **Förvalt installeras programmet per användare utan administratörsbehörighet.** Installern erbjuder även installation för alla användare om den startas med administratörsrätt. Organisationen behöver därför själv spärra installation om bara vissa ska få använda programmet. Tyst central utrullning, även per dator, beskrivs i IT-drift. MSI finns inte.
- **Licensnyckeln är enda faktor på webben.** Nedladdningen av installern på privera.se/download/ och kundsidan Mitt abonnemang, som öppnar Stripes kundportal med kort, kvitton och uppsägning, nås med licensnyckeln och ingen annan faktor. Kundsidan kontrollerar inte om licensen gäller. Den som har nyckeln kan alltså säga upp abonnemanget. Installern lämnas också ut mot köpets sessions-id från bekräftelsesidan efter köpet, om licensen har rätt att köra programmet.

Kräver organisationen MFA eller roller inuti programmet eller på kundsidan uppfyller Privera inte den delen av kravet i dag.

För Itha AB:s egna administrativa konton (Cloudflare, Stripe, Azure, GitHub och e-post) är det inte dokumenterat vilka som har MFA påslaget (Planerat: kontrolleras och dokumenteras).

6. Licensnyckeln i klartext: rättat

Två vägar där licensnyckeln kunde ligga i klartext är rättade.

6.1 Licensfilen på datorn (från 0.9.0-beta20)

Går licensen inte att kryptera med Windows krypteringsskydd skrivs ingenting som bär nyckel eller licensbevis. Licensen gäller då bara tills programmet stängs, och programmet visar att nästa start kan kräva ny aktivering. En befintlig krypterad fil lämnas orörd. En klartextfil från en äldre version krypteras vid start och ersätts.

Före rättningen (0.9.0-beta19 och äldre) skrevs nyckel och licensbevis okrypterade i användarens profil när skyddet saknades.

Kvar: på en dator där krypteringsskyddet saknas vid start ligger en klartextfil från en äldre version kvar tills skyddet fungerar igen. Programmet raderar den inte, eftersom kunden då förlorar licensen. Den ersatta filen skrivs över via namnbyte, inte säker radering, så gamla diskblock kan finnas kvar tills de skrivs över.

6.2 Licensnyckeln i licensdatabasen

För att kunden ska kunna se nyckeln på bekräftelsesidan efter köpet, också efter en omladdning, sparas den tills licensen har aktiverats eller i högst sju dagar. Den sparas krypterad med AES-256-GCM, med en nyckel härledd (HKDF-SHA-256) ur en hemlighet i licensservern och med köpets session-id som bunden tilläggsdata. En kopia av databasen räcker alltså inte för att läsa nyckeln. Saknas hemligheten sparar licensservern ingen nyckel alls. En timkörning tömmer nycklar som är äldre än sju dagar eller vars licens har aktiverats.

Före rättningen låg nyckeln okrypterad i licensdatabasen och raderades bara när kunden återvände till bekräftelsesidan.

Kvar: den som kan köra kod i licensservern, eller har både databasen och hemligheten, kan läsa nycklar under hämtningsfönstret.

7. Övrigt som saknas

Rättat: licens.privera.se skickar HSTS och de övriga säkerhetshuvudena (avsnitt 2), och från 0.9.0-beta22 kräver uppdateringens utgivarkontroll hela utgivarnamnet, avvisar paket för webbinstallation och är stängd vid fel, se IT-drift avsnitt 5.

- privera.se på webbläsarnas förinstallerade HSTS-lista.
- Licensnyckeln och kassans session-id står i adressen vid nedladdningen av installern och vid hämtningen av nyckeln efter köpet. Adressen skyddas av TLS men kan sparas i webbläsarens historik och i loggar. Därför är Cloudflares lagring av licensserverns loggar avstängd (avsnitt 3.2).
- Uppdateringsmanifestet är inte signerat. Det skyddas av TLS, av att bara licenser med rätt att köra programmet får hämta det och, från 0.9.0-beta22, av att den hämtade installerns signatur och version kontrolleras innan den installeras.
- Ingen skriftlig rutin för byte av signeringsnycklar.
- Ingen inställning för krypteringsnivå efter informationsklass, ingen MFA och inga roller i programmet eller på kundsidan.

Underleverantörer

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/underleverantorer/

Redovisar vilka underleverantörer som behandlar uppgifter om Itha AB:s kunder, var uppgifterna finns och på vilken grund de förs över, samt leveranskedjan för själva programmet.

1. Utgångspunkt

Privera behandlar organisationens dokument bara på organisationens egna datorer. Programmet skickar inte dokument, filnamn, träffar eller användningsstatistik till Itha AB eller till någon underleverantör. Dokumenten lagras och behandlas därför bara där organisationens datorer finns.

Undantaget är när organisationen själv väljer att lämna ett dokument till Itha AB för att ett fel ska kunna undersökas. Det sker bara efter ett särskilt personuppgiftsbiträdesavtal för ärendet, och dokumentet raderas senast 30 dagar efter att det tagits emot, om organisationen inte skriftligen begär förlängning. Skickas dokumentet med e-post lagras det som en krypterad arkivfil hos one.com, utan lösenordet. Kopior av arkivfilen kan finnas kvar i one.com:s säkerhetskopior efter raderingen, så länge one.com sparar dem. one.com:s personuppgiftsbiträdesavtal anger ingen tid.

Underleverantörerna nedan behandlar i stället uppgifter om köp, licens, support och drift, eller ingår i kedjan som bygger och levererar programmet.

2. Underleverantörer som behandlar uppgifter om kunden

Underleverantör	Syfte	Uppgifter	Plats	Överföringsgrund	Avtal
Cloudflare, Inc. (USA)	Licensserver (Workers), licensdatabas (D1), lagring av installer och uppdateringar (R2), webbplatsen privera.se (Pages), DNS	D1: e-postadress, SHA-256-summa och fyra sista tecken av licensnyckeln, Stripes kund- och prenumerations-id, maskin-id, datornamn, tidsstämplar, ångerärenden och IP-adress som nycklad pseudonym. R2: bara installationsfiler. Trafikuppgifter (IP-adress) vid varje anrop. Licensserverns säkerhetslogg och driftlogg lagras inte hos Cloudflare i dag: lagringen av loggar (Workers Logs) är avstängd, se Säkerhetsloggning avsnitt 11.	D1 och R2 i region Österuropa (EEUR). Ingen EU-jurisdiktion är satt på någon av dem (kontrollerat 2026-10-05). Workers, Pages och DNS körs i Cloudflares globala nät.	EU-U.S. Data Privacy Framework (DPF) och EU:s standardavtalsklausuler (SCC) i Cloudflares personuppgiftsbiträdesavtal	Cloudflares Customer DPA, som ingår genom hänvisning i Cloudflares villkor
Stripe Payments Europe, Ltd. (Irland)	Kassa, prenumerationer, fakturor och kundportal	Namn, e-post, faktureringsadress och betaluppgifter. Kortuppgifter når aldrig Itha AB.	Irland. Uppgifter kan föras över till Stripe, LLC i USA och till Stripes underbiträden.	SCC genom Stripes Data Transfer Addendum. Stripe uppger att den också kan stödja sig på DPF.	Stripes DPA, som ingår genom hänvisning i Stripe Services Agreement

Underleverantör	Syfte	Uppgifter	Plats	Överföringsgrund	Avtal
one.com	E-post för privera.se, bland annat support@, info@ och feedback@	Mejl som kunden skickar: supportärenden, uppsägningar, ångermeddelanden och säkerhetsrapporter. Vid felsökning enligt särskilt avtal även en krypterad arkivfil, utan lösenordet.	Lagring i datacenter i Köpenhamn (Danmark)	one.com:s personuppgiftsbiträdesavtal (version 17 juni 2025) räknar upp underbiträdena per produkt och nämner inte e-post som egen produkt. För webbhotellet anges datacenter i Danmark och systemövervakning i Danmark och i Indien (One.com India). Underbiträden i USA anges för andra produkter, bland annat webbplatsbyggaren. Vid överföring utanför EU/EES ska one.com enligt avtalet se till att det finns lämpliga skyddsåtgärder, till exempel EU:s standardavtalsklausuler.	one.com:s DPA, som enligt avtalet gäller från samma tidpunkt som huvudavtalet. Att det gäller Itha AB:s konto och vem som är avtalspart är inte bekräftat.
Vercel Inc. (USA)	Itha AB:s interna verktyg: kundregister, säljutskick och driftlarm från licensservern	Organisationsnamn, organisationsnummer och anteckningar om kunder och tänkbara kunder, namn och e-post till kontaktpersoner, och driftlarm som kan innehålla Stripes kund- och prenumerations-id	Funktioner i region Frankfurt	SCC i Vercels DPA	Vercels DPA. Att det gäller Itha AB:s abonnemang är inte bekräftat.
Neon (Databricks, USA)	Databas för samma interna verktyg	Samma som raden ovan	AWS eu-central-1 (Frankfurt)	Inte fastställd	Att ett DPA är ingånget är inte bekräftat.

Statistiken som licensservern lämnar till det interna verktyget är aggregerad och innehåller inga e-postadresser eller id:n.

3. Tjänst som programmet kontaktar direkt

Tjänst	Syfte	Uppgifter	Plats	Överföringsgrund	Avtal
Hugging Face, Inc. (USA; EU-etablering Hugging Face SAS, Frankrike)	Programmet hämtar språkmodellen KB/bert-base-swedish-cased-ner i en låst revision första gången den behövs. Från 0.9.0-beta22 kontaktas Hugging Face inte när modellen redan finns på datorn.	Datorns IP-adress och bibliotekens tekniska anropsuppgifter. Inget dokumentinnehåll.	Hugging Face uppger att uppgifter kan lagras i USA eller andra länder	Organisationens dator kontaktar Hugging Face direkt, och Itha AB för inte över något. Hugging Face anger ingen överföringsmekanism i sin integritetspolicy (kontrollerat 2026-10-05).	Inget avtal mellan Itha AB och Hugging Face. Hugging Face är självständigt ansvarig för sina loggar.

Vill organisationen undvika kontakten kan flödet spärras efter första hämtningen. Att förinstallera modellen centralt stöds inte i dag (Ej infört).

4. Leveranskedjan för programmet

Ingen av dessa har tillgång till organisationens data eller till kunduppgifter.

Led	Leverantör	Roll	Kontroll hos Itha AB
Kodsignering	Microsoft (Azure Trusted Signing, region North Europe), utgivare Itha AB	Signerar installer och uppdateringar. Certifikatets nyckel lämnas aldrig ut av tjänsten.	Releasebygget kräver signering.
Källkod och CI	GitHub (Microsoft)	Privat källkodsförråd. Tester och bygge när koden ändras.	Inga kunduppgifter i förrådet.
Programfönster	Electron 44.6.0 (Chromium) via npm	Programmets fönster	Låst i låsfil med integritetshashar.
Python-paket	PyPI	Motorns bibliotek, bland annat PyTorch	Exakta versioner i låsfil. Hashar saknas (Planerat). Från 0.9.0-beta22 innehåller motorn bara de paket som står i motorns låsfil för körning. Bygget kontrollerar det, och ett signerat bygge vägrar en motor som byggts med oincheckade ändringar.
OCR	Tesseract 5.4.1 och 24 C-bibliotek från conda-forge	Textigenkänning i skannade sidor	Låsta på exakt fil med SHA-256. Bygget stoppas vid GPL-licens eller fel version.
Språkmodell	KBLab vid Kungliga biblioteket, via Hugging Face	Igenkänning av namn och organisationer	Låst revision. Byte kräver ny utvärdering.

En komponentförteckning (SBOM) i CycloneDX 1.5 publiceras för 0.9.0-beta23, se SBOM. Från 0.9.0-beta22 tas den fram ur det byggda paketet i bygget. Den har kända luckor. GitHubs larm om sårbara beroenden (Dependabot) är påslagna sedan 2026-10-23 för npm-beroendena. Motorns Python-paket och OCR-motorns bibliotek omfattas inte, och larmen ger inga automatiska rättningar och körs inte i CI (Planerat).

5. Interna verktyg utanför produkten

Itha AB använder också verktyg för försäljning och administration som inte ingår i Privera men kan beröra kontaktuppgifter:

Verktyg	Uppgifter	Anmärkning
Telegram	Avisering till Itha AB om säljutskick: organisationernas namn och e-postadresser i utskicket, och när en organisation svarar, eller en adress som Itha AB bevakar skickar e-post till den inkorg som säljutskicken skickas från, avsändarens namn eller adress, ämnesrad och upp till 1 500 tecken av texten. Telegram används inte för driftlarm; licensserverns larm går till det interna verktyget hos Vercel och Neon.	Inget personuppgiftsbiträdesavtal. Användningen ska bedömas eller ersättas.

6. Säkerhetskrav bakåt i kedjan

- Med samtliga leverantörer (Cloudflare, Stripe, one.com, Vercel, Neon, Microsoft, GitHub och Telegram) har Itha AB leverantörens standardvillkor. Med Hugging Face har Itha AB inget avtal. Itha AB kan inte avtalsvägen föra över kundens egna krav till dem, och kan därför inte visa att samma säkerhetskrav som kunden ställer gäller längre bak i kedjan. Vi kan i stället redovisa deras avtal och publicerade säkerhetsintyg.
- I utkastet till avtalsbilaga om säkerhet åtar sig Itha AB att meddela byte av en underleverantör som behandlar uppgifter om kunden 30 dagar i förväg till kunder med avtal. Byter eller lägger one.com till egna underbiträden vidarebefordrar Itha AB one.com:s besked till kunder med felsökningsavtal så snart det kommer, vilket kan ge kortare varsel.
- Itha AB är inte certifierat enligt ISO 27001 eller SOC 2. Hur bolagets rutiner förhåller sig till kontrollerna i ISO 27001 står i Kontroller enligt ISO 27001.

Säkerhetsintyg hos de underleverantörer som har licens- och betaluppgifterna (kontrollerat 2026-10-13 mot leverantörernas egna sidor):

Underleverantör	Intyg	Hur rapporterna lämnas ut
Cloudflare	ISO/IEC 27001:2022 för den globala molnplattformen, ISO/IEC 27018 och ISO/IEC 27701. SOC 2 Type II-rapport för säkerhet, konfidentialitet och tillgänglighet, som omfattar Workers, D1 och R2.	Rapporter och certifikat lämnas ut av Cloudflare under sekretessavtal. Källa: cloudflare.com/trust-hub/compliance-resources
Stripe	PCI DSS, tjänsteleverantör nivå 1. SOC 1- och SOC 2 Type II-rapporter som tas fram årligen. Offentlig SOC 3-rapport.	SOC 1 och SOC 2 lämnas ut av Stripe på begäran. SOC 3 är offentlig. Källa: docs.stripe.com/security

7. Det som återstår

- **Garanterad lagring inom EU hos Cloudflare.** Regionerna för D1 och R2 är platsangivelser, inte garantier. Cloudflare beskriver en platsangivelse som "best effort", och en jurisdiktion kan bara sättas när databasen skapas. För att garantera lagring inom EU måste licensdatabasen och lagringen skapas om med jurisdiktionen EU och datan flyttas (Ej infört).
- Bekräfta att personuppgiftsbiträdesavtal gäller för one.com, Vercel och Neon, vem som är avtalspart hos one.com och överföringsgrunden för Neon.
- Bedöma Telegram-flödet.
- Integritetspolicyn på webben nämner inte säljutskicken och Telegram-flödet.
- Automatisk bevakning av sårbarheter i motorns Python-paket och OCR-motorns bibliotek, och licensservern i SBOM:en.
- Lagring av licensserverns loggar hos Cloudflare, när licensnyckeln och session-id:t inte längre står i adressen.

Livscykel och support

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/livscykel-och-support/

Anger hur länge Privera får support och säkerhetsuppdateringar, hur snabbt sårbarheter åtgärdas, hur uppdateringar levereras och vad som händer när produkten når sitt slut. Där något är ett åtagande som inte är byggt står det uttryckligen.

1. Sammanfattning

Fråga	Svar
Supportperiod	Säkerhetsrättningar i 5 år från utgivningen av version 1.0. Slutdatum (månad och år) anges när 1.0 ges ut. Version 1.0 är inte utgiven och har inget fastställt datum.
Under betan	Säkerhetsrättningar fram till att version 1.0 ges ut. Därefter gäller femårsperioden. Uppdateringar och support ingår så länge abonnemanget löper.
Versioner som stöds	Den senaste utgivna versionen. Den ingår i abonnemanget utan extra kostnad.
Åtgärdstid för sårbarheter	Kritisk 7 dagar, Hög 30 dagar, Medel 90 dagar, Låg senast 180 dagar. Fastställda 2026-10-23 (avsnitt 4).
Varsel före supportens slut	Minst 6 månader.
Om Itha AB upphör	Kunden får en slutversion som fungerar utan licensservern. Mekanismen finns från version 0.9.0-beta20 (avsnitt 8).
Utan kontakt med licensservern	Programmet fungerar i högst 14 dagar efter senaste lyckade licensförnyelse, om inget avvecklingsbesked gäller.
Kontakt	Support: support@privera.se. Säkerhet: feedback@privera.se. Dataskyddsfrågor: info@privera.se. Telefon 076 926 35 70.

2. Supportperiod

Supportperioden beslutades 2026-10-23.

Från version 1.0. Itha AB ger säkerhetsrättningar för Privera i 5 år räknat från den dag version 1.0 ges ut. Datumet för 1.0, och därmed supportperiodens slutdatum, anges i månad och år när 1.0 ges ut, på privera.se och i avtalet. Version 1.0 är inte utgiven, och något datum för den är inte fastställt.

Under betan. Privera är i dag en beta före 1.0. Den aktuella versionen är 0.9.0-beta23. Betan får säkerhetsrättningar fram till att version 1.0 ges ut, och därefter gäller femårsperioden ovan. I övrigt gäller köpvillkoren: uppdateringar och support ingår så länge abonnemanget löper, och e-postsupport ges helgfria vardagar klockan 09 till 18.

Förhållande till CRA. EU:s cyberresiliensförordning, förordning (EU) 2024/2847, kräver i artikel 13.8 en supportperiod på minst fem år, om produkten inte förväntas användas kortare tid, och i artikel 13.19 att slutdatumet anges vid köpet. Kraven gäller från 2027-12-11. Fem år från version 1.0 motsvarar huvudregeln i artikel 13.8. Hur perioden ska räknas för ett program som säljs löpande, från version 1.0 eller från varje senare försäljning, har vi inte låtit bedöma juridiskt. Artikel 13.9 kräver dessutom att varje utgiven säkerhetsuppdatering finns tillgänglig i minst 10 år efter utgivningen, eller resten av supportperioden om den är längre. I dag lämnar uppdateringskanalen och nedladdningen bara ut den senaste installern, och äldre versioner finns inte tillgängliga för kunderna. Planerat före 2027-12-11: att varje utgiven säkerhetsuppdatering hålls tillgänglig så som artikel 13.9 kräver. Hur det ska ordnas är inte bestämt.

3. Säkerhetsuppdateringar

- **Vilken version.** Säkerhetsrättningar ges ut i en ny version av programmet. Bara den senaste versionen får rättningar. Äldre versioner rättas inte; kunden uppdaterar till den senaste, som ingår i abonnemanget.
- **Kostnad.** Säkerhetsuppdateringar ingår i abonnemanget utan extra kostnad.

- **Vem som får dem.** Uppdateringar, även säkerhetsrättningar, lämnas bara ut till licenser som har rätt att köra programmet: provperiod, aktivt abonnemang, uppsagt abonnemang till periodens slut och misslyckad kortdragning medan betalningen försöks igen, i samtliga fall till slutet av den betalda perioden eller provperioden plus 3 dagar. Det är samma regel som aktiveringen och licensförnyelsen använder. En uppsagd, obetald, pausad eller utgången licens får inga uppdateringar. Det följer av köpvillkoren (uppdateringar ingår så länge abonnemanget löper) och av att en sådan installation inte kan bearbeta dokument: utan giltigt licensbevis visar programmet bara licensrutan (avsnitt 6). Den som tecknar abonnemanget igen får den senaste versionen vid nästa start.
- **Funktions- och säkerhetsändringar.** I dag levereras säkerhetsrättningar i samma versioner som andra ändringar. När en rättning inte kan vänta på nästa planerade version ges en separat version ut.
- **Tredjepartskomponenter.** Electron och Chromium, Python-paket, Tesseract med bibliotek och språkmodellen omfattas. Larm från GitHub om sårbara beroenden (Dependabot) bevakar npm-beroendena sedan 2026-10-23. Python-paketen och Tesseract med bibliotek bevakas manuellt. Programmet levereras med Electron 44.6.0. Electron ger säkerhetsrättningar för de tre senaste stabila serierna; enligt Electrons tidsplan upphör stödet för 44-serien 2027-03-02, och programmet behöver byta serie före dess, se Säker utveckling.
- **Meddelanden.** Rättade sårbarheter av grad Medel eller högre redovisas i säkerhetsmeddelanden, se Sårbarhetshantering.

4. Åtgärdstider

Tiderna är fastställda 2026-10-23 och är desamma som i Sårbarhetshantering, där graderna också definieras. De räknas från att bedömningen av sårbarheten är klar. Första bedömning görs inom 5 arbetsdagar från mottagandet. Åtgärd betyder en rättad version i uppdateringskanalen eller, om det inte går i tid, en publicerad skyddsåtgärd.

Grad	Åtgärd senast
Kritisk	7 dagar
Hög	30 dagar
Medel	90 dagar
Låg	Nästa planerade version, senast 180 dagar
Aktivt utnyttjad	Snarast. Råd om skyddsåtgärd till berörda kunder senast i uppdateringen 72 timmar efter kännedom, räknat under helgfria vardagar, se Incidentrutin avsnitt 5.

Kan en tidsram inte hållas meddelas berörda kunder med skäl och ny tidpunkt (för Kritisk och Hög). Aktivt utnyttjade sårbarheter och allvarliga incidenter rapporteras enligt CRA artikel 14.

Inkorgen bevakas helgfria vardagar. Det finns ingen jour dygnet runt.

5. Uppdateringskanal

- Programmet frågar licens.privera.se efter en ny version när det startar. Frågan skickas bara om programmet har en licensnyckel.
- Servern lämnar ut uppdateringen, och installern via privera.se/download/, bara till en licensnyckel vars licens har rätt att köra programmet (avsnitt 3). Övriga får samma svar som en okänd nyckel.
- Uppdateringen laddas ner i bakgrunden. Programmet kontrollerar filens SHA-512 och signaturen innan den installeras. Från 0.9.0-beta22 krävs Itha AB:s hela utgivarnamn och den version som uppdateringsmanifestet angav, och kan äktheten inte kontrolleras installeras uppdateringen inte, se IT-drift avsnitt 5.
- Uppdateringen installeras när användaren klickar på beskedet i programmet, eller vid nästa normala avslut. Omstart tvingas aldrig fram.
- Programmet installeras förvalt per användare, utan administratörsbehörighet. Installern kan också köras tyst och per dator, se IT-drift.

Central styrning (från 0.9.0-beta21). Organisationen kan styra uppdateringarna med grupprincip, Intune eller registret: stänga av dem och rulla ut nya versioner själv, eller låta programmet bara meddela att en ny version finns. En ADMX-mall finns. Detaljerna står i IT-drift avsnitt 5 och 6. Med uppdateringarna avstängda ansvarar organisationen för att rulla ut säkerhetsrättningarna, som bara ges i den senaste versionen (avsnitt 3).

Saknas i dag: MSI-paket.

6. Drift utan licensservern

- Licensservern utfärdar ett signerat licensbevis som gäller i högst 14 dagar, och aldrig längre än till slutet av den betalda perioden plus 3 dagar.
- Programmet börjar förnya beviset när halva giltighetstiden har gått, alltså normalt efter 7 dagar, och försöker därefter en gång i timmen.
- När beviset har gått ut slutar programmet bearbeta dokument. Det kontrolleras både i gränssnittet och i programmets huvudprocess.

Följden: om licensservern inte går att nå fungerar programmet i högst 14 dagar efter senaste lyckade förnyelse, om inte ett avvecklingsbesked har lästs in (avsnitt 8). Beviset gäller aldrig längre än till slutet av den betalda perioden eller provperioden plus 3 dagar. Försvinner servern under de sista dagarna av en betalperiod kan det därför bli bara några dagar kvar, i värsta fall cirka 3. Marginalen på 3 dagar förlänger aldrig tiden utan nät; den kan bara korta den.

7. Slut på support

För en version. Rättningar ges i den senaste versionen (avsnitt 3). Det enda som krävs av kunden är att uppdatera till den senaste versionen, som ingår i abonnemanget. Varslet nedan gäller supporten för produkten, inte enskilda versioner.

För produkten. Itha AB meddelar kunderna minst 6 månader innan supporten för Privera upphör. Meddelandet skickas med e-post till kundens registrerade adress och till kontaktpersonen i avtalet. Under varselperioden ges säkerhetsuppdateringar som vanligt.

Saknas i dag: besked om supportens slut inne i programmet. Det är planerat men inte byggt.

8. Om Itha AB upphör eller slutar med Privera

Åtagande. Om Itha AB upphör med verksamheten, försätts i konkurs eller slutar tillhandahålla Privera får kunderna en slutversion av programmet som fungerar utan licensservern. Slutversionen lämnas ut via uppdateringskanalen medan den finns kvar, och som installer direkt till kunden. Det gäller även kunder vars abonnemang redan är uppsagt: när beskedet är publicerat lämnar licensservern ut slutversionen till varje känd licensnyckel. Beskedet publiceras och slutversionen läggs ut innan abonnemangen avslutas.

Hur. Itha AB utfärdar ett avvecklingsbesked som signeras med en egen Ed25519-nyckel, avvecklingsnyckeln. Den är skild från licensserverns nyckel och finns inte i licensservern. Den privata nyckeln ska flyttas till förvaring offline; per 2026-10-23 är det inte gjort. Programmet tar emot beskedet från licensservern vid en förnyelse, eller som fil under Inställningar eller på licensskärmen. Med ett giltigt besked kör programmet utan tidsgräns, utan licenskontroller och utan uppdateringskontroller. Besked som är falska, ändrade eller signerade med fel nyckel avvisas. Se Avvecklingsläge.

Läget i dag. Mekanismen finns från version 0.9.0-beta20. Versioner till och med 0.9.0-beta19 saknar den och slutar bearbeta dokument inom högst 14 dagar utan licensservern (avsnitt 6). Åtagandet gäller därför kunder som har uppdaterat till 0.9.0-beta20 eller senare. Läget är provat i automatiska tester av programmets licensdel och av licensservern, men inte i ett installerat program på en ren dator.

Källkodsdeposition (escrow) erbjuds inte i dag.

Kundens data påverkas inte. Dokumenten behandlas bara på kundens dator. Original, exporterade filer, exportloggar och säkerhetsloggar ligger kvar hos kunden oavsett vad som händer med Itha AB eller licensservern. Avinstallation tar inte bort programmets datamapp.

9. Kontakt

Ärende	Kontakt
Support	support@privera.se, helgfria vardagar klockan 09 till 18
Säkerhet och sårbarheter	feedback@privera.se
Dataskyddsfrågor	info@privera.se
Telefon	076 926 35 70 (besvaras i mån av möjlighet)
Post	Itha AB, Bränningestrandsvägen 40b, 151 39 Södertälje

Säker utveckling

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/saker-utveckling/

Beskriver hur Privera utvecklas, testas, byggs och släpps, och vad som ännu saknas. Det som står under "Finns" är infört. Det som står under "Saknas" eller "Planerat" är inte infört. Datumen i planen är mål, inte avtalade åtaganden.

1. Produkten i korthet

Privera är ett Windows-program som maskerar och avidentifierar dokument. Det består av tre delar:

- ett programfönster byggt med Electron,
- en Python-motor som paketeras med PyInstaller och körs som en separat process, med Tesseract 5.4.1 för OCR och språkmodellen KB/bert-base-swedish-cased-ner,
- en licens- och uppdateringsserver på Cloudflare Workers med D1-databas och R2-lagring.

Dokumenten behandlas bara på användarens dator. Programfönstret och motorn talar med varandra över lokala rör (stdio), inte över en nätverksport. Programmet kontaktar licens.privera.se för licens och uppdateringar och, första gången språkmodellen behövs, Hugging Face.

2. Översikt

Område	Läge
Versionshantering i Git, automatiska tester när koden ändras	Finns
Releasekontroller som stoppar bygget (paketkontroll, spärr mot kända GPL-bibliotek och AGPL, OCR-version, och från 0.9.0-beta22 kontroll av det byggda paketet)	Finns, avsnitt 4
Kodsignering av installer och alla programfiler (Azure Trusted Signing)	Finns
Härdning av programfönstret (Electron)	Finns, med kvarvarande brister (avsnitt 7). Electrons spärrar i programfilen är satta från 0.9.0-beta22.
Regressionstest för varje bekräftat exportfel, samlad körning före varje version	Finns
Mätning av detektionens träffsäkerhet mot en låst baslinje	Finns, körs lokalt
Versionslåsta beroenden	Finns, med luckor (avsnitt 6)
Skriftlig hotmodell	Saknas
Oberoende kodgranskning	Saknas, avsnitt 5
Automatisk bevakning av sårbarheter i beroenden	Delvis. Larm från GitHub (Dependabot) för npm-beroendena sedan 2026-10-23, inte för Python-paketen och inte i CI (avsnitt 6).
SBOM per release	Finns från 0.9.0-beta22: tas fram ur det byggda paketet i bygget. Har kända luckor (avsnitt 6).
Penetrationstest, internt eller externt	Ej genomfört. Offerter tas in under hösten 2026, avsnitt 9.
CI som spärrar release, och grenskydd	Saknas
Fullständig signaturkontroll i den automatiska uppdateringen	Finns från 0.9.0-beta22, avsnitt 8

3. Utveckling

Finns

- All kod versionshanteras i Git. Större ändringar görs på egna grenar och slås ihop.

- Säkerhetskritiska regler är skrivna som kod och tester, inte bara som rutin. Exempel: programmets gränssnitt får inte välja var exportloggen skrivs, huvudprocessen kontrollerar licensen innan motorn får arbeta oberoende av vad gränssnittet visar, och sökvägar från gränssnittet valideras.
- Exportens och detektionens regler är nedskrivna i interna dokument.

Saknas

- En skriftlig hotmodell för hela produkten. Hot har analyserats per område i interna granskningar men inte samlats.
- Ett utvecklingsteam med flera granskare. Nästan alla ändringar görs av en och samma utvecklare.

4. Tester och kontroller

Finns

- **CI vid push och pull request.** CI körs när ändringar skickas till källkodsförrådet, utom till arbetsgrenen, vars ändringar kontrolleras när de når huvudgrenen. En körning gäller den senaste ändringen i varje push. För programfönstret, licensservern och webben körs tester, typkontroll och bygge. Motorns del körs när motorn eller en fil som dess tester läser ändras, och kan startas för hand. Då installeras beroendena från låsfil, OCR-motorn och språkdata hämtas låsta, motorn paketeras med PyInstaller, hela testsviten körs mot källkoden, och en uppsättning obligatoriska regressionstester körs mot den paketerade motorn. Därefter kontrolleras paketet. CI har bara läsrätt till källkodsförrådet.
- **Regressionstester för exportfel.** Varje bekräftat fel där en uppgift överlevde exporten har ett eget regressionstest. Testerna körs i CI mot källkoden. Mot det paketerade programmet körs de lokalt på byggdatorn.
- **Samlad körning av exportfelen** före varje publicering. Ett test som saknas, hoppas över eller kraschar räknas som omätt och ger felkod, så att ett uteblivet mätvärde aldrig ser ut som godkänt.
- **Mätning av träffsäkerhet.** Detektionens precision och träffsäkerhet per kategori mäts mot en låst baslinje, och ändringar som försämrar den fälls. Att flytta baslinjen kräver en skriftlig motivering. Mätningen körs lokalt före release, eftersom utvärderingsmaterialet inte lämnar Itha AB:s dator.
- **Paketkontroll före bygge.** Bygget stoppas om den paketerade motorn eller OCR-motorn saknas, kända GPL-bibliotek (jbig, lzo2) finns bland OCR-motorns filer, OCR-motorn inte är exakt Tesseract 5.4.1 med Leptonica 1.83.1, den paketerade motorn inte matchar källkodens SHA-256, språkdata saknas eller ett AGPL-beroende finns, eller avvecklingsnyckeln saknas, inte är en Ed25519-nyckel eller är samma som licensserverns nyckel (kontrolleras i det signerade kundbygget).
- **Kontroll av det byggda paketet** (från 0.9.0-beta22). Efter att programmet har packats, innan installern skapas och igen efter bygget, stoppas bygget om Electrons spärrar i programfilen (fuses) har fel läge, programfilens binära version inte är den som hör till versionen eller inte är högre än den senast utgivna, installationskatalogen har programfiler som inte kommer från Electron, eller en programfil har en ändelse som döljer den, `app.asar` inte innehåller den kod som byggdes, eller de uppackade filerna bredvid den inte stämmer med den, motorn innehåller paket som inte står i dess låsfil för körning, eller motorn i paketet inte är den som byggdes. Ett signerat kundbygge stoppas också om källkoden eller motorn har oincheckade ändringar, om en programfil saknar signatur, om något av NSIS-tilläggen i installern eller avinstalleraren inte är giltigt signerat, om installern inte klarar programmets egen signaturkontroll eller om SBOM:en inte kan tas fram.
- **Licensserverns driftsättning** stoppas om driftdatabasen saknar migreringar.
- **Versionskonsekvens.** Ett test i CI fallerar om installerns namn i licensservern inte stämmer med programmets version.

Saknas

- CI spärrar inte release. Installern byggs, signeras och publiceras från Itha AB:s byggdator, och det finns inget tekniskt krav att CI är grön innan dess. Grenskydd med obligatorisk CI är inte infört och går inte att slå på med det GitHub-abonnemang som källkodsförrådet har i dag, ett privat förråd utan betalt abonnemang.
- CI har inte varit grön vid varje utgivning. Alla körningar 2026-10-07, dagen då 0.9.0-beta20 och 0.9.0-beta21 gavs ut, föll. Enligt vår felsökning berodde det på två fel i testerna som bara uppträdde på GitHubs Windows-maskin,

inte i programmet, men skrivbordsdelens typkontroll och bygge kördes inte i de körningarna. Av versionerna 0.9.0-beta14 till 0.9.0-beta21 gavs bara 0.9.0-beta19 ut medan CI var grön. CI är grön sedan 2026-10-13.
- Mätningen av träffsäkerhet och den samlade körningen av exportfelen startas för hand. Resultaten är dokumenterade för en del versioner men inte för alla.

5. Kodgranskning

Finns

- Säkerhetsrättningar granskas internt i två steg: en granskning per område och därefter en motgranskning där felen återskapas med körda prov. Granskningarna görs av Itha AB självt, med verktyg som bolaget styr, och är inte oberoende.

Saknas

- Oberoende kodgranskning. Ingen ändring har formellt granskats av en annan person: ingen av de 17 pull requests som har slagits ihop har någon granskning, och övriga ändringar har slagits ihop utan pull request. Ingen extern part har granskat koden.
- Krav på granskning före sammanslagning. Grenskydd går inte att slå på med förrådets nuvarande GitHub-abonnemang.

Planerat

- Mål Q1 2027: en extern säkerhetsgranskning av koden för export, uppdatering och licens.
- Mål Q1 2027: regel om att ändringar i export, uppdatering, licens och signering granskas av en andra person före release.

6. Tredjepartskomponenter

Finns

- Python-paketen är låsta på exakt version för Windows x64 och Python 3.12.10. Låsfilen får inte uppdateras utan att träffsäkerheten mäts.
- npm-beroendena är låsta i låsfil med integritetshashar.
- Tesseract och dess C-bibliotek hämtas från conda-forge, låsta på exakt fil och kontrollerade mot SHA-256. Hämtningen kontrollerar paketens licenstexter och stoppar GPL-licensierade bibliotek.
- Språkmodellen är låst till en bestämd revision.
- Spärr mot kända GPL-bibliotek i OCR-motorn och mot AGPL-beroenden i motorn. Electron-delen och Python-delen kontrolleras inte mot GPL.
- Från 0.9.0-beta22 innehåller motorn bara de paket som står i dess låsfil för körning, och bygget stoppas annars (avsnitt 4). I 0.9.0-beta21 följde utvecklingsverktygen pytest och mypy och biblioteket zstandard med i motorn.
- SBOM i CycloneDX 1.5, från 0.9.0-beta22 framtagen ur det byggda paketet i bygget. Se SBOM.
- Larm från GitHub om sårbara beroenden (Dependabot), påslagna sedan 2026-10-23. De gäller npm-beroendena. Larmen ger inga automatiska rättningar och körs inte i CI.

Saknas

- Automatisk bevakning av motorns Python-paket och OCR-motorns bibliotek, och kontroll av beroenden i CI: varken npm audit, pip-audit eller OSV körs, och Dependabot-larmen gäller bara npm-beroendena. Sårbarheter i övriga beroenden upptäcks i dag bara vid manuell genomgång.
- Licensservern i SBOM:en.
- Hashlåsning av Python-paketen och låsning av GitHub Actions på commit i stället för tagg.
- En uppdateringstakt för Electron. Programmet levereras med Electron 44.6.0, en serie som får säkerhetsrättningar. Enligt Electrons tidsplan upphör stödet för den 2027-03-02. Någon fast rutin för att byta serie i tid finns inte.

Planerat (datumen är förslag, se avsnitt 11)

- Mål 2026-11-15: npm audit, pip-audit och Dependabot i CI.
- Mål 2026-11-30: licensservern i SBOM:en.
- Mål 2026-11-30: hashlåsning av Python-paketen och låsning av Actions på commit.

7. Härdning av programmet

Finns

- `contextIsolation: true`, `nodeIntegration: false` och `sandbox: true` för programfönstret.
- Nya fönster nekas. Externa länkar öppnas bara i webbläsaren om adressen finns på en tillåtelselista.
- Navigering bort från programmet blockeras.
- Content Security Policy: inga externa skript, `connect-src 'self'`, `object-src 'none'`, `frame-src 'none'`. Inline-stilar är tillåtna.
- Licensserverns adress kan inte pekas om med miljövariabler i kundversionen.
- Licensbevis signeras med Ed25519 och kontrolleras mot en publik nyckel som är inbyggd i programmet. Den privata nyckeln finns inte i källkoden.
- Avvecklingsbeskedet signeras med en egen Ed25519-nyckel som är skild från licensservern. En angripare som tar över licensservern kan inte låsa upp programmet för alltid. Den privata nyckeln ska flyttas till förvaring offline; per 2026-10-23 är det inte gjort.
- Sparade granskningar och licensfilen skyddas med Windows DPAPI via Electron safeStorage. Ingen av dem skrivs i klartext om skyddet saknas; för licensfilen gäller det från 0.9.0-beta20.
- Licensnyckeln som väntar på bekräftelsesidan efter köpet förseglas med AES-256-GCM under en hemlighet i licensservern.
- Electrons inbyggda spärrar i programfilen (fuses), från 0.9.0-beta22: körning som Node (`RunAsNode`), `NODE_OPTIONS` och `--inspect` är avstängda, och asar-integritet, `OnlyLoadAppFromAsar` och kryptering av kakor är påslagna. Bygget stoppas om en spärr har fel läge. Se IT-drift avsnitt 4.1.
- Felsökningsporten (`--remote-debugging-port` med flera) avslutar kundutgåvan innan något fönster öppnas, utom när organisationen tillåter den med policyn `AllowDebugPort`, från 0.9.0-beta22.
- Windows egna program som programmet startar, PowerShell och `reg.exe`, startas med full sökväg i Windows-katalogen och med en egen miljö, från 0.9.0-beta22. Katalogen tas ur den plats Windows laddade `ntdll.dll` från, inte ur miljövariabeln `SystemRoot`.

Brister vi känner till

- Spärren `GrantFileProtocolExtraPrivileges` står kvar på, eftersom programfönstret laddas från en fil med `file:` och inte laddas alls utan den. Att stänga av den kräver att fönstret laddas genom ett eget protokoll.
- Asar-integriteten skyddar bara `app.asar`. I en installation per användare kan användaren byta ut DLL-filer, `.node`-filer bredvid `app.asar` och motorn. Det som stoppar det är AppLocker eller WDAC, se IT-drift avsnitt 4.
- I 0.9.0-beta21 och äldre är spärrarna inte satta. Den signerade programfilen i de versionerna kan användas för att köra annan kod än programmets egen, och en regel i AppLocker eller WDAC som tillåter Itha AB:s signatur släpper in dem. AppLocker behöver därför en nekande regel för de versionerna, se IT-drift avsnitt 4.

8. Bygge, signering och leverans

Finns

- Installern byggs med tvingande kodsignering, så ett osignerat bygge går inte att få ut som installer.
- Signering sker via Microsoft Azure Trusted Signing med utgivaren Itha AB.
- Varje programfil i paketet (.exe, .dll, .pyd, .node) signeras om den saknar signatur. Bygget stoppas om någon programfil saknar giltig signatur eller om en av Itha AB:s signaturer saknar tidsstämpel.

- Från 0.9.0-beta22 signeras också installerns och avinstallerarens NSIS-tillägg (bland annat `System.dll` och `nsis7z.dll`), som packas upp och laddas från `%TEMP%` under installationen. I 0.9.0-beta21 och äldre var de osignerade, och på en dator med Smart App Control påslaget installerades programmet då tyst utan programfiler: installern gav kod 0 men ingen `Privera.exe`. Se IT-drift avsnitt 2.1.
- Installer och uppdateringar lämnas bara ut till licenser som har rätt att köra programmet.
- Licensservern skickar HSTS och övriga säkerhetshuvuden på varje svar, och skriver en säkerhetslogg utan licensnyckel, e-post och IP-adress i klartext. Loggen sparas inte i dag. Se Kryptering och åtkomstkontroll avsnitt 2 och Säkerhetsloggning avsnitt 11.
- Programfilen och installern har en binär filversion som stiger med varje version, 0.9.0.22 för 0.9.0-beta22. Bygget stoppas om den inte är högre än den senast utgivna. Det gör att en AppLocker-regel kan neka 0.9.0-beta21 och äldre, se IT-drift avsnitt 4.
- Installern läggs inte ut för nedladdning utan en SBOM från samma signerade bygge.
- Den automatiska uppdateringen kontrollerar filens SHA-512 mot uppdateringsmanifestet före installation och tvingar aldrig fram en omstart. Från 0.9.0-beta22 har programmet en egen signaturkontroll: signaturen ska ha status Valid och gälla den hämtade filen, utgivarens namn ska ha alla delarna CN, O, L, S och C med Itha AB:s värden, och filens versionsuppgift ska vara den som manifestet angav; en fil utan läsbar versionsuppgift avvisas. Kontrollen är stängd vid fel, paket för webbinstallation godtas inte, och ett nej loggas (5003). Kan äktheten inte kontrolleras på datorn, till exempel för att PowerShell är spärrat, installeras inga uppdateringar, och Om Privera säger det. Se IT-drift avsnitt 5.

Brister vi känner till

- Uppdateringsmanifestet är inte signerat. Det skyddas av TLS, av att bara licenser med rätt att köra programmet får hämta det och av att installerns signatur och version kontrolleras innan den installeras.
- Går den hämtade installerns versionsuppgift inte att läsa avgörs uppdateringen på signaturen.
- Ett uppdateringsfel som inte gäller äktheten, till exempel ett nätverksfel, syns inte för användaren, bara som `uppdatering.fel` (5002) i säkerhetsloggen.
- I 0.9.0-beta21 och äldre jämför signaturkontrollen bara certifikatets namn (CN), och där Windows PowerShell inte kan köras hoppas den över. En dator som uppdaterar från 0.9.0-beta21 prövar 0.9.0-beta22 med den äldre kontrollen.

9. Säkerhetstestning

Finns

- Riktad testning av exporten: försök att få påhittade personuppgifter att överleva en export. Varje bekräftat fel blir ett regressionstest (avsnitt 4).

Saknas

- Penetrationstest. Inget penetrationstest av programmet (kommunikationen mellan processer, uppdateringsflödet, filhantering, motorprocessen) eller av licensservern (licens.privera.se, betalkopplingen, databasen) har gjorts, varken internt eller av extern part.

Planerat

- Offerter på ett externt penetrationstest av skrivbordsappen och licensservern tas in under hösten 2026; målet är ett genomfört test under Q1 2027. När ett test är gjort har kunder som har undertecknat säkerhetsbilagan rätt att ta del av det. Övriga kunder kan begära en sammanfattning.

10. Sårbarheter, incidenter och livscykel

Hur sårbarheter tas emot, bedöms och åtgärdas beskrivs i Sårbarhetshantering. Supportperiod, åtgärdstider och vad som gäller vid avveckling beskrivs i Livscykel och support.

Sedan 2026-09-11 omfattas Itha AB som tillverkare av rapporteringsplikten i EU:s cyberresiliensförordning, förordning (EU) 2024/2847, artikel 14.

11. Planen samlad

Datumen är förslag och inte beslutade. När de är beslutade gäller samma datum i Plan för CRA.

Åtgärd	Mål (förslag)	Läge
Fullständig signaturkontroll i uppdateringen, del 1: fullt utgivarnamn, avstängda webbinstallerare, synligt besked (avsnitt 8)	2026-10-31	Klart, i 0.9.0-beta22
Automatisk bevakning av beroenden i CI (npm audit, pip-audit, Dependabot)	2026-11-15	Delvis. Dependabot-larm för npm-beroendena sedan 2026-10-23. Inte i CI och inte för Python-paketen: Planerat.
SBOM framtagen automatiskt per release och kontrollerad mot bygget	2026-11-30	Klart, i 0.9.0-beta22. Licensservern i SBOM:en: Planerat.
Release byggs bara från en version med grön CI	Ej fastställt	Förslag, ej beslutat
Electron som får säkerhetsrättningar (Electron 44)	2026-11-30	Klart
Hashlåsning, låsning av Actions på commit	2026-11-30	Planerat
Skriftlig hotmodell	Q1 2027	Planerat
Granskning av en andra person för säkerhetskritiska ändringar	Q1 2027	Planerat
Externt penetrationstest av skrivbordsappen och licensservern	Offerter hösten 2026, genomfört test Q1 2027	Planerat
Extern kodgranskning	Q1 2027	Planerat, budget ej fastställd
Electrons fuses satta i programfilen (avsnitt 7)	Ej fastställt	Klart, i 0.9.0-beta22. `GrantFileProtocolExtraPrivileges` står kvar på.
Egen signaturkontroll i uppdateringen, del 2: avvisar uppdateringen när kontrollen inte går att genomföra	Q1 2027	Klart, i 0.9.0-beta22
Teknisk dokumentation och EU-försäkran enligt CRA	Senast 2027-12-11	Planerat

Itha AB är inte certifierat enligt ISO 27001 eller SOC 2.

SBOM

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/sbom/

Beskriver komponentförteckningen (Software Bill of Materials) för Privera: format, innehåll, hur den tas fram och vilka luckor den har.

1. Var den finns

- SBOM:en för version 0.9.0-beta23, den aktuella versionen, finns på privera.se/security/sbom/privera-0.9.0-beta23.cdx.json, med kontrollsumma (SHA-256) i privera-0.9.0-beta23.cdx.json.sha256.
- Äldre versioner: SBOM:erna för 0.9.0-beta20, 0.9.0-beta21 och 0.9.0-beta22 finns på samma plats, och den för 0.9.0-beta19 lämnas ut på begäran. SBOM:erna för 0.9.0-beta19 till 0.9.0-beta21 togs fram på det äldre sättet och har luckorna i avsnitt 7. Frågor om SBOM:en skickas till feedback@privera.se.
- SBOM:en är inte signerad. Kontrollsumman visar att filen inte har ändrats sedan den skrevs, men inte vem som skrev den.
- Från 0.9.0-beta22 tas SBOM:en fram ur det byggda paketet, i samma bygge som installern (avsnitt 4). Den har ändå kända luckor (avsnitt 6).

2. Format

- CycloneDX 1.5, JSON.
- Varje komponent har typ och namn, och varje komponent utom filerna har purl eller CPE. CPE anges där namnet är känt. Nästan alla komponenter har version (avsnitt 5).
- Varje programfil (.exe, .dll, .pyd och .node) står som filkomponent under den komponent den hör till, med SHA-256, storlek, versionsuppgift och om den bär en Authenticode-signatur.
- Beroendegrafen är med.
- serialNumber räknas fram ur innehållet, så två körningar med samma indata ger samma nummer.
- Metadatan anger vilken version av källkoden programmet och motorn är byggda ur, om arbetskopian hade ändringar som inte var incheckade, om det är ett signerat kundbygge, antalet programfiler och installerns filnamn och SHA-256.
- Komponenterna har egenskaper som anger var i programmet de ligger (`privera:paketering`) och varifrån version och licens kommer.

3. Vad SBOM:en täcker

Från 0.9.0-beta22 läses uppgifterna ur det paket som byggts, inte ur låsfilerna. En komponent som inte finns i paketet kommer inte med, och en programfil i paketet som inte kan knytas till någon komponent stoppar bygget.

Del	Med i SBOM:en
Programfiler	Varje programfil i paketet, med SHA-256 och signaturuppgift, under sin ägare. Avinstalleraren skapas först vid installationen och har därför ingen hash i SBOM:en.
npm-paket	Paketen i programmet, med version, purl och SHA-512 ur låsfilen, och de paket som är inbyggda i programfönstrets kodpaket.
Electron	Electron samt versionerna av Chromium, Node.js och V8 som ingår, prövade mot programfilen.
Installerare	NSIS 3.0.4.1 med sina insticksprogram och `elevate.exe`. Installerns egen SHA-256 står i metadatan.
Python-motorn	De paket som faktiskt finns i den paketerade motorn, med version, purl och licens. Motorns binärfiler är jämförda byte för byte med paketens egna filförteckningar, före signeringen. CPython 3.12.10 och PyInstaller, vars startprogram ingår. Itha AB:s egen motor, med källkodens kontrollsumma.

Del	Med i SBOM:en
C-bibliotek i motorn	Med version och källa, bland annat OpenSSL 3.0.16, SQLite 3.49.1, zlib, expat, libffi, bzip2 och xz, som följer med CPython, libxml2 2.11.9 och libxslt 1.1.45 (i lxml), libyaml 0.2.5 (i PyYAML), qpdf med en statiskt länkad OpenSSL (i pikepdf), PDFium (i pypdfium2), OpenBLAS (i numpy) och libuv, Intel MKL och Intel OpenMP (i PyTorch).
Microsofts C- och C++-runtime	En komponent per version som följer med, med filerna under sig.
Inbäddade SBOM:er	Tokenizers, safetensors, Pillow, hf_xet och pydantic_core har egna SBOM:er, bland annat över sina Rust-delar. Deras komponenter står med, och SBOM-filerna refereras med SHA-256. HarfBuzz och raqm är inbyggda i Pillows typsnittsmodul och står med som ingående, med en notering att de är inaktiva eftersom FriBiDi saknas vid körning. Delar som Pillows SBOM nämner men vars kod inte finns i någon fil i paketet (FriBiDi, libimagequant och libxcb) är märkta `excluded`.
Tesseract och C-bibliotek	Tesseract 5.4.1 och de paket från conda-forge som bidrar med filer, med version, build, kanal och SHA-256. Varje fil i OCR-motorn är knuten till sitt paket och jämförd byte för byte med det.
Språkdata	`swe.traineddata` och `eng.traineddata` med SHA-256.
Språkmodellen	KB/bert-base-swedish-cased-ner vid en låst revision, som `machine-learning-model`, med MIT-licens och viktfilens SHA-256. Modellen ingår inte i installern utan hämtas första gången den behövs.

Några uppgifter finns inte i paketet och står därför i skriptet tillsammans med sin källa: vilka versioner av Chromium, Node.js och V8 som hör till Electron-versionen, hashen för modellens viktfil enligt Hugging Face, licensen för språkmodellen och språkdatan och PyInstallers licens. Uppgifterna gäller bara exakt den version som står angiven, och Electron-uppgifterna prövas mot programfilen. Byts Electron-versionen eller modellens revision stoppar skriptet. Byts språkdatans version skrivs ingen licens in för den, och byts PyInstallers version anges dess licens utan undantaget för startprogrammet.

4. Så tas den fram och kontrolleras

- Från 0.9.0-beta22 tas SBOM:en fram av bygget självt, direkt efter att paketet och installern är byggda. Den skrivs bara om kontrollerna av det byggda paketet har gått igenom: Electrons spärrar i programfilen, programfilens version, att motorn bara innehåller paketen i sin låsfil och är den motor som byggdes, och programfilernas signaturer.
- Ett signerat kundbygge vägrar källkod med oincheckade ändringar och en motor som byggts med oincheckade ändringar, och varje programfil måste bära en Authenticode-signatur.
- SBOM:en prövas mot den officiella schemafilen för CycloneDX 1.5 innan den skrivs, och mot egna krav: varje id förekommer en gång, varje komponent har purl eller CPE och beroendegrafen hänger ihop. SBOM:en från provpaketeringen av 0.9.0-beta22 godkändes också av två andra verktyg för CycloneDX, och en medvetet förstörd kopia fälldes av alla tre.
- Installern läggs inte ut för nedladdning utan en SBOM från ett signerat bygge som anger just den installerns SHA-256.
- SBOM:erna för 0.9.0-beta19 till 0.9.0-beta21 togs fram för hand efter bygget, ur låsfilerna (avsnitt 7).

5. Komponenter med ofullständiga uppgifter

Utan version eller licens. Några komponenter som kommer ur de inbäddade SBOM:erna saknar version eller licens, eftersom paketets egen SBOM inte anger den.

Språkmodellens licens. Språkmodellen KB/bert-base-swedish-cased-ner har MIT-licens. KBLab vid Kungliga biblioteket angav licensen på modellkortet för KBLab/bert-base-swedish-cased-ner den 15 oktober 2026 och bekräftade den skriftligt till Itha AB samma dag. Filerna som Privera laddar är byte för byte desamma i de båda förråden. Från 0.9.0-beta22 anger SBOM:en licensen, och licenstexten följer med installationen. SBOM:erna för 0.9.0-beta21 och äldre togs fram före beskedet och anger ingen licens för modellen.

Licens som fritext i stället för SPDX-id. Uppgiften finns men går inte att maskinläsa entydigt, till exempel "BSD License" utan variant eller "Apache Software License" utan version. Det gäller bland annat colorama, jinja2, mpmath, sympy, safetensors, tokenizers och pypdfium2, några av conda-paketen, Microsofts runtime, Intels bibliotek samt Itha AB:s egen motor, som är proprietär.

Rättad licens. PyInstaller har licensen satt i skriptet i stället för ur paketets metadata, och bara för en angiven version: GPL-2.0-or-later WITH Bootloader-exception, som tillåter att startprogrammet sprids inbäddat.

6. Kända luckor

- **Ingen sårbarhetsskanning mot SBOM:en**, till exempel med OSV eller Grype. Flera av de inbyggda biblioteken är äldre versioner. OpenSSL 3.0.16, som följer med CPython i motorn, har kända sårbarheter. Motorn använder det för den krypterade förbindelsen när språkmodellen hämtas från Hugging Face första gången, och motorn gör inga andra nätverksanrop. Se Sårbarhetshantering.
- **Inga hashar för Python-paketen som paket.** Programfilerna har SHA-256, och motorns binärfiler är jämförda med paketens filförteckningar, men paketen har ingen egen hash, eftersom låsfilen saknar hashar. Python-kod som ligger packad i motorns arkiv har ingen hash i SBOM:en.
- **Chromiums tredjepartskomponenter listas inte som egna komponenter.** Licenserna finns i `LICENSES.chromium.html` i installationen.
- **Kod som byggverktygen lägger in listas inte**, till exempel hjälpkod i gränssnittets kodpaket och grundstilar.
- **Licensservern ingår inte än.** Licensservern levereras inte till kunden, men räknas som produktens fjärrdatabehandling enligt CRA och ska därför in i SBOM:en (Planerat, mål 2026-11-30). Webbplatsen ingår inte i produkten.

7. Äldre SBOM:er (0.9.0-beta19 till 0.9.0-beta21)

De togs fram för hand efter bygget, ur de filer som styr bygget, främst låsfilerna, och skiljer sig därför från det som paketerades. En jämförelse 2026-10-23 av SBOM:en för 0.9.0-beta21 med den signerade, paketerade 0.9.0-beta21 visade följande. De publicerade filerna är inte ändrade i efterhand.

- **Paket i motorn som inte stod i SBOM:en:** utvecklingsverktygen pytest och mypy med sina beroenden (pluggy, iniconfig, mypy_extensions, librt och ast_serialize), och biblioteket zstandard. Från 0.9.0-beta22 följer de inte med i motorn.
- **Paket i SBOM:en som inte fanns i motorn:** cffi, cryptography, pdfminer-six, pdfplumber och pycparser.
- **Källkodsversionen:** motorn i 0.9.0-beta21 byggdes ur en närliggande version med ändringar som inte var incheckade. Skillnaden gällde enligt vår kontroll bara byggfiler, men det går inte att bevisa i efterhand.
- **Inbäddade bibliotek saknades som egna komponenter**, bland annat OpenSSL 3.0.16 och SQLite i CPython, PDFium, qpdf, motorns C- och C++-runtime från Microsoft och Rust-paketen i tokenizers. SBOM:ens OpenSSL-post gällde en annan version, den som följer med Tesseract.
- **Alla låsta conda-paket stod med**, också de som inte bidrar med någon fil till installern.
- `nsis` saknade version, och språkmodellen saknade licens.

8. CRA

EU:s cyberresiliensförordning, förordning (EU) 2024/2847, bilaga I del II punkt 1, kräver att tillverkaren upprättar en SBOM i ett vanligt maskinläsbart format som täcker minst produktens direkta beroenden. Förordningen gäller fullt ut från 2027-12-11. Om CRA också kräver att SBOM:en lämnas till kunden har vi inte kontrollerat. Vi publicerar den ändå.

Avvecklingsläge

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/avvecklingslage/

Beskriver hur Privera kan fortsätta fungera utan licensservern om Itha AB upphör eller slutar med Privera: åtagandet, hur mekanismen fungerar och i vilken ordning ett avvecklingsbesked utfärdas.

1. Åtagande

Om Itha AB upphör med verksamheten, försätts i konkurs eller slutar tillhandahålla Privera får kunderna en slutversion av programmet som fungerar utan licensservern. Det gäller även kunder vars abonnemang redan är uppsagt. Åtagandet står också i Livscykel och support.

2. Läget

Del	Läge
Mekanismen i programmet (verifiering, avvecklingsläge, inläsning från fil och från licensservern)	Finns från version 0.9.0-beta20.
Licensserverns stöd (utdelning av beskedet, slutversionen till varje känd licens)	Byggt och testat.
Avvecklingsnyckeln	Skapad 2026-10-07. 0.9.0-beta20 är den första version som bär den. Den privata nyckeln är skild från licensservern. Den ska flyttas till förvaring offline; per 2026-10-23 är det inte gjort, och nyckeln ligger på en av Itha AB:s datorer.
Versioner till och med 0.9.0-beta19	Saknar mekanismen. De slutar bearbeta dokument senast 14 dagar efter senaste lyckade licensförnyelse om licensservern försvinner. Åtagandet gäller kunder som har uppdaterat till 0.9.0-beta20 eller senare.
Provat i ett installerat program på en ren dator	Nej. Provat i automatiska tester av programmets licensdel och av licensservern (avsnitt 7).

3. Hur det fungerar

Två nycklar, med flit

Licensservern signerar licensbevis med en nyckel som finns i licensservern. Ett licensbevis gäller i högst 14 dagar.

Ett avvecklingsbesked gäller utan tidsgräns och på varje dator som får det. Det signeras därför med en egen Ed25519-nyckel, avvecklingsnyckeln, som är skild från licensservern och inte finns där. Den som tar över licensservern kan alltså inte utfärda ett besked som programmet godtar. Bygget stoppas om avvecklingsnyckeln är samma som licensserverns nyckel.

Beskedets format

```
PRVA1.<base64url(JSON)>.<base64url(ed25519-signatur)>
```

Signaturen täcker `PRVA1.<base64url(JSON)>`, alltså även prefixet. Ett licensbevis (prefix `PRVL1`) kan därför inte läsas som ett avvecklingsbesked. Nyttolasten:

```
{ "v": 1, "typ": "avveckling", "utfardat": <unix-s>, "giltigFran": <unix-s>,
  "meddelande": "<text till kunderna, högst 2000 tecken>" }
```

Beskedet är inte knutet till någon kund eller dator. Det är ett offentligt besked till alla kunder och skyddas av signaturen, inte av att hållas hemligt.

Vad programmet gör

- **Verifiering.** Signaturen prövas mot den publika avvecklingsnyckeln som är inbyggd i programmet. Programmet avvisar ett besked med fel nyckel, ändrat innehåll, ändrad signatur, fel typ, fel version, saknade fält eller fel prefix.
- **Två vägar in.**
  1. *Från licensservern.* När programmet ändå ska förnya licensen frågar det först licensservern efter ett besked. Anropet har ingen kropp, ingen licensnyckel och inget maskin-id, och sker bara vid förnyelser, så det blir inga nya kontakttillfällen. Licensservern visar bara upp beskedet; programmet prövar signaturen själv.
  2. *Från fil.* Användaren väljer "Läs in avvecklingsbesked" under Inställningar, Licens, eller på licensskärmen om programmet redan har låst sig. En fil över 64 KB läses inte. Filen får innehålla förklarande text runt beskedet; bara den signerade raden räknas.
- **Lagring.** Ett godkänt besked sparas i programmets datamapp. Signaturen prövas igen vid varje start. Ändras eller byts filen ut blir svaret "inget besked", inte ett fel.
- **När beskedet gäller** (från `giltigFran`):
  - Programmet får köra, oavsett licensbevis, dator och klocka.
  - Programmet gör inga licensanrop. Förnyelse, aktivering, frånkoppling och kundportalen avböjs lokalt med ett besked på svenska.
  - Programmet gör ingen uppdateringskontroll. Det kommer inga fler versioner, och domänen kan en dag hamna hos någon annan.
  - Inställningar visar att Privera körs i avvecklingsläge och fungerar utan licensserver, och visar Itha AB:s meddelande.
- **Innan beskedet gäller** fungerar licensen som vanligt, och Inställningar visar när beskedet börjar gälla. En klocka som ställs bakåt efter giltighetsdagen slår inte av läget. Att ställa klockan framåt för att nå läget tidigare ger bara det som Itha AB redan har lovat ut.

4. Ordningen när beskedet utfärdas

Itha AB åtar sig att följa ordningen nedan. Om åtagandet också binder en konkursförvaltare är inte juridiskt bedömt; rutinen är skriven så att en förvaltare kan följa den. Beskedet publiceras och slutversionen läggs ut innan abonnemangen avslutas. Sägs abonnemangen upp först nekar licensservern nästa förnyelse, och programmet slutar bearbeta dokument direkt i stället för efter 14 dagar.

1. Itha AB bestämmer från vilket datum beskedet ska gälla. Det kan vara samma dag. Datum tolkas som midnatt UTC.
2. Beskedet signeras med avvecklingsnyckeln på en dator som inte är ansluten till nätet, och provas i ett installerat program innan det skickas ut.
3. Den senaste versionen med mekanismen läggs ut som slutversion i uppdateringskanalen och nedladdningen. Kunder på 0.9.0-beta19 eller äldre får mekanismen genom att uppdatera till den.
4. Beskedet publiceras i licensservern. Från och med då:
   - delar licensservern ut beskedet, och program med mekanismen hämtar det vid nästa förnyelse, normalt inom 7 dagar,
   - lämnar uppdateringskanalen och nedladdningen på privera.se/download/ ut slutversionen till varje känd licens, också uppsagda och obetalda. En påhittad nyckel nekas som förut. Det som lämnas ut är programmet, inte rätten att köra det: utan besked eller giltigt licensbevis bearbetar programmet inga dokument,
   - svarar licensservern en licens utan rätt med ett tillfälligt fel i stället för ett definitivt nej vid förnyelse. Programmet behåller då sitt licensbevis tills det går ut, också i 0.9.0-beta19 och äldre, och hinner hämta slutversionen under tiden.
5. Beskedet skickas som fil med e-post till varje kunds registrerade adress och till kontaktpersonen i avtalet, också till kunder som redan har sagt upp, och läggs ut på privera.se så länge domänen finns.
6. Först därefter avslutas abonnemangen, tidigast samma dag som steg 4 och helst efter en vecka, när de flesta program har hunnit fråga.

7. Licensservern hålls kvar i minst 14 dagar efter steg 6, så att program som bara har ett licensbevis kvar hinner hämta slutversionen.

Ett besked kan inte dras tillbaka.

Om ordningen ändå bryts och abonnemangen sägs upp före steg 4: program som redan har låst sig gör ingen ny förnyelse och hämtar därför inte beskedet från servern. Uppdateringskontrollen görs däremot vid varje start med den sparade nyckeln, så när beskedet väl är publicerat hämtar även ett låst program slutversionen. Beskedet läses sedan in som fil från licensskärmen.

5. Risker

Risk	Följd	Skydd
Den privata avvecklingsnyckeln läcker	Vem som helst kan låsa upp alla installationer utan tidsgräns. Det kostar Itha AB intäkter, men kunddata röjs inte: beskedet ger bara rätt att köra programmet lokalt. Det kan inte återkallas i redan utgivna versioner.	Nyckeln hålls skild från licensservern. Förvaring offline är planerad men inte gjord per 2026-10-23. Vid läcka byts nyckeln i en ny version.
Licensservern eller dess hemligheter blir angripna	Angriparen kan visa upp ett falskt besked, men programmet godtar det inte.	Separat nyckel som inte finns i licensservern. Licensservern signerar ingenting här.
Avvecklingsnyckeln går förlorad	Åtagandet kan inte uppfyllas för versioner med den nyckeln.	Planerad förvaring offline, inte gjord per 2026-10-23. Går nyckeln förlorad ges en ny version med ett nytt nyckelpar ut medan Itha AB finns, och kunderna behöver uppdatera.
Kunden har en version utan mekanismen (0.9.0-beta19 eller äldre)	Programmet låser sig senast 14 dagar efter att servern har försvunnit.	Kunden uppdaterar till en version med mekanismen. Slutversionen kan också delas ut som installer.
Abonnemangen sägs upp innan beskedet är publicerat	Programmet låser sig vid nästa förnyelse, och uppsagda licenser får ingen slutversion förrän beskedet är publicerat.	Ordningen i avsnitt 4. Ett låst program hämtar ändå slutversionen när beskedet är publicerat, och beskedet kan läsas in som fil.
Domänen tas över efter avvecklingen	En främling kan svara på licens.privera.se.	I avvecklingsläge gör programmet inga licens- eller uppdateringsanrop.

6. Byte av nyckel

Den publika avvecklingsnyckeln är inbyggd i programmet. Ett byte kräver en ny version, och äldre versioner godtar fortfarande bara den gamla nyckeln. Den gamla privata nyckeln behålls därför så länge versioner med den kan vara i bruk. Är två nycklar i bruk utfärdas ett besked med vardera nyckeln, i två filer.

7. Tester

Mekanismen är provad i automatiska tester:

- Verifieringen: giltigt besked, fel nyckel, ändrat innehåll, ändrad signatur, fel typ, fel version, saknade fält, licensbevis signerat med samma nyckel, skräp, giltighetsdag och bakåtställd klocka.
- Licensdelen mot riktig disk utan licensserver: programmet låser sig utan besked och fungerar ett år senare med besked, inga nätanrop görs i avvecklingsläge, inläsning från fil överlever omstart, ett felsignerat eller manipulerat besked avvisas, läget väntar in giltighetsdagen, beskedet hämtas från servern vid en förnyelse och ett falskt besked från servern avvisas.
- Ett besked från Itha AB:s skript godtas av programmet, och ett besked med en annan nyckel godtas inte. Byggspärren stoppar en nyckel med fel form och en återanvänd licensnyckel.
- Filinläsningen går via programmets egen fildialog, och en för stor fil läses inte. Frågan efter beskedet har ingen kropp och ingen nyckel.

- Licensservern: inget svar utan besked, beskedet utan krav på nyckel, uppsagda och obetalda licenser nekas uppdateringar och installer utan besked men släpps fram med besked, en okänd nyckel nekas fortfarande, och förnyelsen av en uppsagd licens ger ett tillfälligt fel i stället för ett nej när beskedet är publicerat.

Läget är inte provat i ett installerat program på en ren dator.

Kontroller enligt ISO 27001

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/iso-27001/

Itha AB är inte certifierat enligt ISO 27001 eller SOC 2. Dokumentet beskriver i stället hur bolagets rutiner förhåller sig till kontrollerna i bilaga A i ISO/IEC 27001:2022, så att organisationen kan jämföra med sina egna krav. Det är Itha AB:s egen bedömning. Ingen extern part har granskat den, och den ersätter inte en certifiering.

Läget anges som **Finns** (infört), **Delvis**, **Saknas** eller **Ej tillämpligt**. Detaljerna står i de dokument som tabellerna hänvisar till.

1. Varför certifieringen väger lättare för Privera

En certifiering visar att en leverantör skyddar de uppgifter den tar emot. Privera är byggt så att Itha AB inte tar emot organisationens dokument vid vanlig användning:

- **Dokumenten stannar hos organisationen.** Analys, maskering och export görs på organisationens egna datorer. Programmet skickar inte dokument, filnamn, träffar eller användningsstatistik till Itha AB eller till någon underleverantör. Undantaget är när organisationen själv väljer att lämna ett dokument för felsökning. Det sker bara efter ett särskilt personuppgiftsbiträdesavtal för ärendet, och dokumentet raderas senast 30 dagar efter att det tagits emot, om organisationen inte skriftligen begär förlängning. Skickas det med e-post kan kopior av den krypterade arkivfilen finnas kvar i e-postleverantörens säkerhetskopior så länge leverantören sparar dem.
- **Det Itha AB har är uppgifter om köp, licens och support.** Licens- och betaluppgifterna ligger hos Cloudflare och Stripe, som har säkerhetsintyg. Cloudflare, som driver licensservern, licensdatabasen och lagringen av installationsfiler, är certifierat enligt ISO/IEC 27001:2022, ISO/IEC 27018 och ISO/IEC 27701 och har en SOC 2 Type II-rapport som omfattar Workers, D1 och R2. Stripe, som hanterar betalningen, är certifierat enligt PCI DSS som tjänsteleverantör på nivå 1 och har SOC 1- och SOC 2 Type II-rapporter samt en offentlig SOC 3-rapport. Supportmejl ligger hos one.com, och Itha AB:s interna kundregister och driftlarm hos Vercel och Neon. För dem redovisar Itha AB inga säkerhetsintyg. Se Underleverantörer.

Det som ändå behöver granskas hos Itha AB är själva programmet: att maskeringen håller, att installer och uppdateringar inte går att manipulera och att licensservern är skyddad. Det är dit tabellerna nedan riktar sig.

2. Organisatoriska kontroller (A.5)

Kontroll	Läge	Beskrivning
5.1 Policyer för informationssäkerhet	Delvis	Rutinerna är beskrivna i dokumenten under /security/it/. En samlad, beslutad säkerhetspolicy finns inte.
5.2 Roller och ansvar	Delvis	En person är incidentansvarig. Ersättare är inte utsedd, se Incidentrutin.
5.3 Åtskillnad av arbetsuppgifter	Saknas	Nästan alla ändringar görs av en utvecklare. Granskning av en andra person för säkerhetskritiska ändringar är planerad till Q1 2027, se Säker utveckling.
5.5 och 5.6 Kontakt med myndigheter och intressegrupper	Finns	Rutin för rapportering till CERT-SE, ENISA och IMY. Inloggningen som krävs för rapportering enligt EU:s cyberresiliensförordning finns.
5.7 Hotunderrättelser	Delvis	Larm från GitHub (Dependabot) om sårbara npm-beroenden sedan 2026-10-23. Motorns Python-paket och OCR-motorns bibliotek bevakas manuellt.
5.9 Förteckning över tillgångar	Delvis	Komponentförteckning (SBOM) publiceras för programmet, från 0.9.0-beta22 framtagen ur det byggda paketet, se SBOM. Licensservern ingår inte i den än. Underleverantörerna är redovisade.
5.14 Överföring av information	Finns	Programmet för inte över några dokument. Vid felsökning enligt särskilt personuppgiftsbiträdesavtal överförs dokumentet via organisationens egen tjänst för säker filöverföring eller som en AES-256-krypterad arkivfil, med lösenordet på annat sätt än med e-post, till exempel per telefon eller sms. Trafiken till licensservern går över TLS 1.2 eller 1.3, se Kryptering och åtkomstkontroll.

Kontroll	Läge	Beskrivning
5.15 till 5.18 Åtkomst, identitet, autentisering och behörigheter	Delvis	Programmet saknar egna konton och följer Windows-kontot, så organisationens inloggning och flerfaktorsautentisering gäller. För Itha AB:s egna administrativa konton är det inte dokumenterat vilka som har flerfaktorsautentisering.
5.19 till 5.22 Leverantörsrelationer	Delvis	Underleverantörer, uppgifter och överföringsgrunder är redovisade. Att personuppgiftsbiträdesavtal gäller för alla är inte bekräftat, se Underleverantörer.
5.23 Molntjänster	Delvis	Licensdatabas och lagring ligger hos Cloudflare i region Östeuropa. Lagring inom EU är inte garanterad.
5.24 till 5.28 Incidenthantering	Delvis	Rutin för att ta emot, bedöma, begränsa, rapportera och följa upp finns. Ersättare och övning av rutinen saknas.
5.29 och 5.30 Kontinuitet	Delvis	Om Itha AB upphör får kunderna en slutversion som fungerar utan licensserver, se Avvecklingsläge. Mekanismen är inte provad i ett installerat program på en ren dator.
5.31 Rättsliga krav	Delvis	Rapportering enligt cyberresiliensförordningen gäller sedan 2026-09-11. CE-märkning planeras före 2027-12-11, se Plan för CRA.
5.34 Integritet och skydd av personuppgifter	Delvis	Dokumenten behandlas bara hos organisationen, utom vid felsökning enligt särskilt personuppgiftsbiträdesavtal. Integritetspolicyn redovisar underleverantörerna för köp, licens och support, men ännu inte behandlingen vid säljutskick. Att personuppgiftsbiträdesavtal gäller för one.com, Vercel och Neon är inte bekräftat.
5.35 Oberoende granskning av informationssäkerheten	Saknas	Ingen extern granskning och inget penetrationstest har gjorts. Offerter på ett externt penetrationstest av skrivbordsappen och licensservern tas in under hösten 2026; målet är ett genomfört test under Q1 2027. Extern kodgranskning är planerad till Q1 2027.
5.37 Dokumenterade driftsrutiner	Finns	Installation, central styrning och avinstallation står i IT-drift.

3. Personalkontroller (A.6)

Kontroll	Läge	Beskrivning
6.1 till 6.7 Bakgrundskontroll, villkor, utbildning, avslut och distansarbete	Saknas	Itha AB är ett litet bolag. Rutiner för detta är inte dokumenterade.
6.8 Rapportering av säkerhetshändelser	Finns	Säkerhetsproblem rapporteras till feedback@privera.se. En security.txt enligt RFC 9116 finns, se Sårbarhetshantering.

4. Fysiska kontroller (A.7)

Kontroll	Läge	Beskrivning
7.1 till 7.14 Lokaler och utrustning	Delvis	Dokumenten behandlas i organisationens egna lokaler och på dess datorer, utom vid felsökning enligt särskilt personuppgiftsbiträdesavtal, då de finns på en namngiven dator hos Itha AB med krypterad disk. Itha AB har inga egna serverlokaler. Licensservern drivs hos Cloudflare, vars datacenter ingår i deras certifiering. Skyddet av Itha AB:s egna datorer, bland annat byggdatorn, är inte dokumenterat.

5. Tekniska kontroller (A.8)

Kontroll	Läge	Beskrivning
8.5 Säker autentisering	Delvis	Licensbevis signeras med Ed25519 och kontrolleras mot en publik nyckel i programmet. För Itha AB:s egna administrativa konton är det inte dokumenterat vilka som har flerfaktorsautentisering, se 5.15 till 5.18.

Kontroll	Läge	Beskrivning
8.7 Skydd mot skadlig kod	Delvis	Installern signeras av Itha AB, och programfiler som saknar signatur signeras vid bygget. Bygget stoppas om någon programfil saknar giltig signatur. Från 0.9.0-beta22 kräver den automatiska uppdateringen Itha AB:s hela utgivarnamn och den version som manifestet angav, och avvisar uppdateringen när kontrollen inte kan köras. Electrons spärrar i programfilen är satta, och programfilens version stiger med varje version, så att AppLocker kan stänga ute äldre versioner. Kvar: i en installation per användare kan användaren byta ut programmets filer, se IT-drift avsnitt 4.
8.8 Hantering av tekniska sårbarheter	Delvis	Bedömning med CVSS och tidsramar för åtgärd finns, se Sårbarhetshantering. Automatisk bevakning finns bara för npm-beroendena (Dependabot-larm sedan 2026-10-23), och ingen sårbarhetsskanning görs mot SBOM:en.
8.9 Konfigurationshantering	Delvis	Uppdateringar och loggning kan styras centralt med registerpolicy och ADMX-mall. Policyn styr vanlig användning men skyddar inte mot en användare som avsiktligt kringgår den, se Säkerhetsloggning avsnitt 7.1.
8.11 Maskering av data	Finns	Produktens huvudfunktion. Vald text tas bort ur filen, inte bara täcks över. Organisationen bedömer själv om uppgifter som finns kvar kan peka ut någon.
8.12 Förhindrande av dataläckage	Finns	Exporten kontrolleras med flera metoder och stoppas om en kontroll underkänns. Varje bekräftat exportfel har ett eget regressionstest.
8.13 Säkerhetskopiering	Delvis	Kundens filer lagras bara hos kunden, utom vid felsökning enligt särskilt personuppgiftsbiträdesavtal. Då säkerhetskopierar Itha AB inte dokumenten, men kopior av den krypterade arkivfilen kan finnas kvar i e-postleverantörens säkerhetskopior. Säkerhetskopiering av licensdatabasen är inte dokumenterad.
8.15 Loggning	Delvis	Säkerhetslogg till Windows händelselogg och hashkedjad fil, som kan samlas in till SIEM. Programmet upptäcker manipulation av loggen men hindrar den inte; det kan bara organisationens behörigheter på loggkatalogen och insamlingen till SIEM. Insamlingen är inte provad mot något SIEM. Licensservern skriver en säkerhetslogg med pseudonymer i stället för licensnyckel och IP-adress, men den sparas inte i dag, se Säkerhetsloggning.
8.20 till 8.22 Nätverkssäkerhet	Finns	Programmet öppnar ingen inkommande port. Utgående trafik går bara till licensservern och, första gången språkmodellen behövs, till Hugging Face.
8.24 Kryptografi	Delvis	TLS 1.2 eller 1.3 i transit, och HSTS på privera.se och licens.privera.se. Sparade granskningar och licensfilen skyddas med Windows DPAPI. En skriftlig rutin för byte av signeringsnycklar saknas.
8.25 till 8.29 Säker utveckling och testning	Delvis	Tester och kontroller när koden ändras och före varje version finns. Skriftlig hotmodell, penetrationstest och spärr som kräver godkända tester före release saknas, se Säker utveckling.
8.28 Säker kodning	Delvis	Programfönstret har isolerat innehåll, sandlåda, blockerad navigering och strikt innehållspolicy. Från 0.9.0-beta22 är Electrons spärrar (fuses) i programfilen satta och felsökningsporten avstängd i kundutgåvan. Spärren `GrantFileProtocolExtraPrivileges` står kvar på, och en skriftlig hotmodell saknas, se Säker utveckling avsnitt 7.
8.32 Ändringshantering	Delvis	All kod versionshanteras och ändringar görs på egna grenar. Granskning av en andra person saknas.

6. Det som saknas, samlat

- Certifiering enligt ISO 27001 eller SOC 2. Itha AB har inget beslut om certifiering.
- Externt penetrationstest: offerter tas in under hösten 2026, och målet är ett genomfört test under Q1 2027. Extern kodgranskning, planerad till Q1 2027. Budget är inte fastställd.
- Information i integritetspolicyn om säljutskicken.
- Dokumentation av flerfaktorsautentisering för Itha AB:s administrativa konton.
- Ersättare för den incidentansvarige, och övning av incidentrutinen. Inkorgen bevakas helgfria vardagar, och det finns ingen jour.
- Samlad säkerhetspolicy, rutiner för personal och dokumenterat skydd av Itha AB:s egna datorer.
- Granskning av en andra person för säkerhetskritiska ändringar, och skriftlig hotmodell.

- Automatisk bevakning av sårbarheter i motorns Python-paket och OCR-motorns bibliotek, kontroll av beroenden i CI och sårbarhetsskanning mot SBOM:en.
- Lagring av licensserverns säkerhetslogg, som kräver att licensnyckeln och kassans session-id först flyttas ur adressen.
- Garanterad lagring inom EU hos Cloudflare.

Datumen är förslag och inte beslutade. De står i Säker utveckling och Plan för CRA.

Plan för CRA

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/cra/

Beskriver Priveras läge mot EU:s cyberresiliensförordning (CRA), förordning (EU) 2024/2847: om produkten omfattas, klassning, rapportering, supportperiod och tidsplanen fram till CE-märkning. Datumen i avsnitt 7 är förslag och inte beslutade, utom det som står som klart. Supportperioden i avsnitt 5 är beslutad.

1. Läget i korthet

- Itha AB kan i dag inte intyga CE-märkning enligt CRA och har ingen EU-försäkran om överensstämmelse för Privera. Ingen bedömning av överensstämmelse är gjord.
- CRA:s huvudkrav, däribland CE-märkning och EU-försäkran, gäller från 2027-12-11. Ingen tillverkare är skyldig att CE-märka enligt CRA före det datumet.
- Rapporteringsplikten i artikel 14 gäller redan, sedan 2026-09-11, och omfattar även produkter som har släppts ut före 2027-12-11.
- Itha AB åtar sig att ha teknisk dokumentation, EU-försäkran och CE-märkning på plats senast 2027-12-11.

2. Omfattas Privera?

Ja. Privera är programvara som tillhandahålls på EU-marknaden i kommersiell verksamhet, mot betalning, och Itha AB marknadsför den under eget namn. Privera är därmed en produkt med digitala element, och Itha AB är tillverkare.

Licensservern licens.privera.se räknas enligt vår läsning som en del av produkten. Definitionen av produkt med digitala element omfattar produktens fjärrdatabehandling, alltså databehandling på distans som tillverkaren har utformat och utan vilken produkten inte kan utföra en av sina funktioner. Utan licensservern går det inte att aktivera eller förnya licensen eller få uppdateringar. Licensservern ingår därför i bedömningen, och därmed också i den tekniska dokumentationen och i SBOM:en. Den nuvarande SBOM:en täcker programmet och motorn. Licensservern läggs in enligt avsnitt 7.

Hämtningen av språkmodellen från Hugging Face är inte Itha AB:s fjärrdatabehandling. Modellen är en tredjepartskomponent och hanteras som en sådan.

Privera är märkt beta men säljs löpande till betalande kunder. Vi stöder oss inte på något undantag för ofärdig programvara.

3. Klassning

Bedömning: Privera är en produkt i standardkategorin, varken viktig (bilaga III) eller kritisk (bilaga IV).

En produkt är viktig om den har kärnfunktionen hos en kategori i bilaga III. Priveras kärnfunktion är att hitta och ta bort personuppgifter ur dokument före utlämnande. Ingen kategori i bilaga III har den funktionen. De kategorier som ligger närmast:

Kategori i bilaga III	Bedömning
Klass I.2, fristående och inbäddade webbläsare	Privera bygger på Electron, som innehåller Chromium. Fönstret visar bara programmets eget gränssnitt och är inte en webbläsare för användaren. Enligt artikel 7.1 gör integrering av en komponent med en kategoris kärnfunktion inte i sig att den större produkten omfattas av bedömningsförfarandena för viktiga produkter.
Klass I.1, identitets- och behörighetshantering	Licenskontrollen avgör om programmet får köras på en dator. Den hanterar inte användares identiteter eller behörigheter.
Klass I.7, SIEM	Exportloggen och säkerhetsloggen registrerar programmets egna händelser så att organisationens SIEM kan samla in dem. Privera samlar inte in eller analyserar andra systems säkerhetshändelser.

Bilaga IV (kritiska produkter) avser hårdvara som säkerhetsmoduler, smartmätargateways och säkra element. Den är inte tillämplig.

Återstår: bedömningen ska prövas mot de tekniska beskrivningarna i kommissionens genomförandeförordning (EU) 2025/2392 och dokumenteras i den tekniska dokumentationen. Det är inte gjort.

Följd: för en produkt i standardkategorin får tillverkaren visa överensstämmelse genom intern kontroll (modul A), utan anmält organ. Det är den väg Itha AB planerar att använda.

4. Det som gäller redan nu: rapportering enligt artikel 14

Sedan 2026-09-11 ska Itha AB som tillverkare rapportera:

Händelse	Tidig varning	Anmälan	Slutrapport
Aktivt utnyttjad sårbarhet i Privera	Inom 24 timmar från kännedom	Inom 72 timmar	Senast 14 dagar efter att en rättning eller skyddsåtgärd finns
Allvarlig incident som påverkar Priveras säkerhet	Inom 24 timmar från kännedom	Inom 72 timmar	Inom en månad efter anmälan

Rapporten lämnas en gång via ENISA:s gemensamma rapporteringsplattform till den CSIRT där tillverkaren har sitt huvudsakliga etableringsställe, för Itha AB CERT-SE. Plattformen kräver ett EU Login-konto med tvåfaktorautentisering. Enligt ENISA:s anvisning registrerar sig tillverkaren när en anmälan ska göras, och CERT-SE prövar behörigheten parallellt med anmälan. Berörda användare ska också informeras.

Läge:

- Rutin: se Incidentrutin.
- EU Login med tvåfaktorautentisering: klart 2026-10-07. Registrering i plattformen görs vid första anmälan, enligt ENISA:s anvisning.
- Kontaktväg för sårbarhetsrapporter: feedback@privera.se och security.txt, se Sårbarhetshantering.
- Möjlighet att nå berörda användare: kontaktuppgifter finns per licens i licensdatabasen. Programmet rapporterar inte sin version, så Itha AB kan inte avgöra vilka kunder som kör en viss version.
- Bemanning: en person, ingen ersättare och ingen jour. Fristen på 24 timmar gäller alla dagar, men Itha AB kan inte garantera att en händelse som inträffar på en helg eller helgdag blir känd och rapporterad inom 24 timmar.

5. Supportperiod

Beslut 2026-10-23: säkerhetsrättningar i 5 år från utgivningen av version 1.0. Betan får säkerhetsrättningar fram till att 1.0 ges ut. Varslet före supportens slut är minst 6 månader. Beslutet ersätter det tidigare beslutet om 24 månader. Se Livscykel och support.

CRA artikel 13.8 kräver att supportperioden är minst fem år, om produkten inte förväntas användas kortare tid. Artikel 13.19 kräver att slutdatumet (minst månad och år) anges tydligt vid köpet, och artikel 13.9 att varje säkerhetsuppdatering finns kvar tillgänglig i minst tio år eller resten av supportperioden, beroende på vilket som är längst.

Läget mot kraven:

- Fem år motsvarar huvudregeln i artikel 13.8. Hur perioden ska räknas för ett program som säljs löpande, från version 1.0 eller från varje senare försäljning, är inte juridiskt bedömt.
- Slutdatumet anges när version 1.0 ges ut. Version 1.0 är inte utgiven och har inget fastställt datum.
- I dag lämnar uppdateringskanalen och nedladdningen bara ut den senaste installern. Planerat före 2027-12-11: att varje utgiven säkerhetsuppdatering hålls tillgänglig så som artikel 13.9 kräver. Hur det ska ordnas är inte bestämt.

6. Väsentliga krav (bilaga I): läget

Översikt, inte en fullständig genomgång. Den fullständiga genomgången blir en del av den tekniska dokumentationen.

Område	Finns i dag	Saknas
Riskbedömning (artikel 13.2)	Säkerhetsöversikt och interna granskningar	Dokumenterad cybersäkerhetsriskbedömning
Inga kända utnyttjbara sårbarheter vid leverans	Regressionstester och samlad körning av exportfelen före release. Electron 44, en serie som får säkerhetsrättningar. Dependabot-larm för npm-beroendena sedan 2026-10-23.	Automatisk bevakning av motorns Python-paket och OCR-motorns bibliotek, kontroll av beroenden i CI (npm audit, pip-audit) och sårbarhetsskanning mot SBOM:en. En fast rutin för att byta Electron-serie innan stödet upphör.
Säker standardkonfiguration	Electron med `contextIsolation` och `sandbox`, utan `nodeIntegration`. Från 0.9.0-beta22 är Electrons spärrar i programfilen (fuses) satta och felsökningsporten avstängd i kundutgåvan, se Säker utveckling avsnitt 7.	Spärren `GrantFileProtocolExtraPrivileges` står kvar på. Genomgång och dokumentation.
Säkerhetsuppdateringar	Signerade uppdateringar som installeras automatiskt vid avslut. Från 0.9.0-beta22 en egen signaturkontroll mot hela utgivaren och manifestets version, stängd vid fel och utan webbinstallerare.	Signerat uppdateringsmanifest
Skydd av data (konfidentialitet)	Lokal behandling, DPAPI för sparade granskningar och licensfil. Licensnyckeln skrivs inte i klartext från 0.9.0-beta20 och sparas förseglad i licensdatabasen.	Se Kryptering och åtkomstkontroll avsnitt 7
Registrering av relevant intern aktivitet	Lokal exportlogg per export. Säkerhetslogg med hashkedja, Windows händelselogg och policystyrd loggplats från 0.9.0-beta20, och från 0.9.0-beta22 analys och påbörjad export (se Säkerhetsloggning). Licensservern skriver en säkerhetslogg med pseudonymer.	Användarinformation om hur loggningen stängs av, lagring av licensserverns säkerhetslogg, samt punkterna under Begränsningar i Säkerhetsloggning
SBOM och sårbarhetshantering (bilaga I del II)	Versionslåsning, OCR-komponenter låsta med SHA-256, SBOM i CycloneDX 1.5, från 0.9.0-beta22 framtagen ur det byggda paketet i bygget	Licensservern i SBOM:en, och sårbarhetsskanning mot den
Information till användaren (bilaga II)	Säkerhetsöversikt, kontaktpunkt för sårbarheter	Supportperiodens slutdatum och länk till EU-försäkran

7. Tidsplan fram till CE-märkning

Senast	Steg	Läge
2026-09-11	Rapporteringsplikten enligt artikel 14 börjar gälla	Gäller
2026-10-07	Kontaktväg för sårbarheter och security.txt publicerade, och EU Login med tvåfaktorautentisering för anmälan i ENISA:s plattform	Klart
2026-10-23	Sårbarhetspolicyn med fastställda tidsramar och supportperioden (avsnitt 5)	Klart
Ej fastställt	Incidentrutinen övad och en ersättare utsedd	Rutinen finns. Övning och ersättare saknas.
Ej fastställt	Säkerhetsbilaga om incidenter och revision för kunder med avtal, juristgranskad	Utkast för juristgranskning

Senast	Steg	Läge
2026-10-31	Fullständig signaturkontroll i uppdateringen, del 1: fullt utgivarnamn, avstängda webbinstallerare, synligt besked när äktheten inte kan kontrolleras	Klart, i 0.9.0-beta22
Ej fastställt	Electrons spärrar (fuses) satta i programfilen	Klart, i 0.9.0-beta22, utom `GrantFileProtocolExtraPrivileges`
2026-11-15	Automatisk bevakning av beroenden i CI (npm audit, pip-audit, Dependabot)	Delvis: Dependabot-larm för npm-beroendena sedan 2026-10-23. I CI och för Python-paketen: Planerat.
2026-11-30	SBOM framtagen automatiskt per release, kontrollerad mot bygget och utökad med licensservern	Framtagen i bygget och kontrollerad mot det: klart, i 0.9.0-beta22. Licensservern: Planerat.
2026-11-30	Byte till en Electron-version som får säkerhetsrättningar	Klart, Electron 44
2026-12-31	Licensnyckeln aldrig i klartext, varken på datorn eller i licensdatabasen	Klart. På datorn från 0.9.0-beta20, i licensdatabasen sedan rättningen i licensservern i oktober 2026.
Q1 2027	Fullständig signaturkontroll i uppdateringen, del 2: egen signaturkontroll	Klart, i 0.9.0-beta22
Q1 2027	Externt penetrationstest av skrivbordsappen och licensservern. Offerter tas in under hösten 2026.	Planerat
2027-03-31	Klassningen dokumenterad mot (EU) 2025/2392, cybersäkerhetsriskbedömning enligt artikel 13.2 och supportperioden i avsnitt 5 prövad mot artikel 13.8	Planerat
2027-06-30	Genomgång mot bilaga I del I och II med åtgärdslista, och bevakning av harmoniserade standarder	Planerat
2027-09-30	Säkerhetsloggning enligt åtgärdslistan och användarinformation enligt bilaga II	Pågår. Första versionen av säkerhetsloggen ingår i 0.9.0-beta20.
2027-10-31	Teknisk dokumentation enligt bilaga VII komplett	Planerat
2027-11-30	Intern kontroll (modul A) genomförd, EU-försäkran (bilaga V) undertecknad, CE-märkning i installer, Om-dialog och på webben	Planerat
2027-12-11	CRA gäller fullt ut	

8. Utfästelse

Itha AB åtar sig att:

1. rapportera enligt CRA artikel 14 inom förordningens frister, räknat från att Itha AB får kännedom, och underrätta berörda kunder. Med en person och utan bevakning på helger kan Itha AB inte garantera att en händelse som inträffar på en helg eller helgdag blir känd inom 24 timmar,
2. ha teknisk dokumentation, EU-försäkran om överensstämmelse och CE-märkning för Privera på plats senast 2027-12-11,
3. redovisa läget mot tidsplanen i avsnitt 7 på kundens begäran.

Stöd för kundens NIS2-efterlevnad

Itha AB · Gäller Privera 0.9.0-beta23 · Senast uppdaterad 2026-11-12 · privera.se/security/it/nis2/

Beskriver hur Privera och Itha AB stöder en kund som omfattas av NIS2-direktivet, direktiv (EU) 2022/2555: vad produkten och Itha AB bidrar med, vad kunden själv behöver göra och vad som saknas i dag. Bilagan går igenom artikel 21.2 punkt för punkt och beskriver också artikel 20 och 23. Den gäller version 0.9.0-beta22 och ersätter inte kundens egen riskbedömning.

1. Utgångspunkt

NIS2-direktivet är genomfört i Sverige genom cybersäkerhetslagen (2025:1506), som trädde i kraft den 15 januari 2026. Kraven i lagen riktar sig mot kunden som verksamhetsutövare. Itha AB berörs som leverantör, främst genom direktivets artikel 21.2 d (säkerhet i leveranskedjan) och 21.3 (kundens bedömning av sina leverantörers sårbarheter och utvecklingsrutiner).

Itha AB bedömer att bolaget inte självt är verksamhetsutövare enligt lagen. Bedömningen är inte juridiskt prövad. Som tillverkare omfattas Itha AB i stället av EU:s cyberresiliensförordning, se Plan för CRA.

Kortfattat är läget detta: Privera har funktioner som stöder kundens arbete, till exempel lokal behandling, säkerhetslogg och central styrning, men flera av de krav som en kund kan ställa på en leverantör är inte uppfyllda i dag. Det står i kolumnen Begränsningar i avsnitt 3 och samlat i avsnitt 7.

2. Produkten i korthet

Egenskap	Uppgift
Produkt	Privera 0.9.0-beta23. Program för Windows 10 och 11 (x64) som maskerar personuppgifter i PDF- och Word-dokument. Version 1.0 är inte utgiven.
Tillverkare	Itha AB, org.nr 559282-3826, Bränningestrandsvägen 40b, 151 39 Södertälje
Behandling av dokument	Lokalt på kundens dator: analys, textigenkänning, språkmodell, maskering och export.
Dokumentdata till Itha AB eller någon annan	Inga dokument, filnamn, träffar eller användningsstatistik. Ingen telemetri och ingen felrapportering. Undantag: ett dokument som kunden själv lämnar för felsökning enligt ett särskilt personuppgiftsbiträdesavtal.
Delar	Programfönster byggt med Electron, en Python-motor som körs som en separat process, Tesseract för OCR och språkmodellen KB/bert-base-swedish-cased-ner.
Licens och uppdatering	Licensservern licens.privera.se hos Cloudflare.
Nätverk	Utgående HTTPS till licens.privera.se och, första gången språkmodellen behövs, till Hugging Face. Ingen inkommande port. Detaljer i Underlag för IT och informationssäkerhet.

3. Artikel 21.2 punkt för punkt

Kolumnen Privera och Itha AB anger vad produkten och leverantören bidrar med. Kolumnen Kunden anger vad som ligger kvar hos kunden.

Art. 21.2	Privera och Itha AB	Kunden	Begränsningar i dag
a) Riskanalys och säkerhetspolicy	Dokumenten lämnar inte datorn, så kundens riskanalys kan avgränsas till arbetsstationen och de två utgående flödena. Arkitektur, flöden och lagring beskrivs i säkerhetsunderlaget.	Klassa informationen, till exempel med KLASSA, och bestämma var original, exporter och loggar får lagras.	Itha AB har ingen dokumenterad cybersäkerhetsriskbedömning enligt CRA artikel 13.2, ingen skriftlig hotmodell och ingen samlad, beslutad säkerhetspolicy.

Art. 21.2	Privera och Itha AB	Kunden	Begränsningar i dag
b) Incidenthantering	Säkerhetslogg från 0.9.0-beta20 till Windows händelselogg och till en hashkedjad fil: start, licens, export, stoppade exporter, läckkontroll, rensning, inställningar och uppdateringar, utan dokumentinnehåll eller filnamn. Från 0.9.0-beta22 står varje export i händelseloggen innan utfilen skrivs, och organisationen kan kräva det med policy. Exportlogg per export med SHA-256 för källfil och utfil. Licensservern skriver en säkerhetslogg med pseudonymer i stället för licensnyckel och IP-adress. Itha AB har en incidentrutin, rapporterar enligt CRA artikel 14 och underrättar berörda kunder (avsnitt 5).	Egen rapportering enligt cybersäkerhetslagen. Insamling av händelseloggen till SIEM, loggkatalog och rensningsförbud med registerpolicy, och append-only-behörighet på loggkatalogen. Se Säkerhetsloggning.	Loggen upptäcker manipulation men hindrar den inte. En vanlig användare kan fylla händelseloggen så att äldre poster skrivs över, och en administratör kan rensa den. I en installation per användare kan användaren ändra programmets filer och därmed kringgå registerpolicyn och skrivningen till händelseloggen. Insamlingen är inte provad med Windows Event Forwarding eller i något SIEM. Licensserverns säkerhetslogg sparas inte och når inte kundens SIEM. Säkerhetsärenden hanteras av en person utan ersättare och utan jour. Incidentrutinen är inte övad. Ingen kund har undertecknat säkerhetsbilagan.
c) Kontinuitet och krishantering	Programmet fungerar utan nät i upp till 14 dagar efter senaste lyckade licensförnyelse. Dokumenten ligger kvar hos kunden oavsett vad som händer med Itha AB. Om Itha AB upphör får kunderna en slutversion som fungerar utan licensserver (avvecklingsläge från 0.9.0-beta20). Säkerhetsrättningar ges till betan fram till version 1.0 och därefter i 5 år från utgivningen av 1.0. Varsel minst 6 månader före supportens slut. Se Livscykel och support.	Säkerhetskopiering av egna dokument och exporter, och en plan för vad som gäller om licensservern inte går att nå.	Avvecklingsläget är inte provat i ett installerat program på en ren dator. Avvecklingsnyckeln ska flyttas till förvaring offline; per 2026-10-23 är det inte gjort. Utan licensserver och utan avvecklingsbesked slutar programmet bearbeta dokument inom högst 14 dagar. Version 1.0 är inte utgiven, så supportperiodens slutdatum är inte fastställt. Bara den senaste versionen får rättningar.
d) Säkerhet i leveranskedjan	Underleverantörer, syfte, uppgifter, land och överföringsgrund redovisas i Underleverantörer. Ingen underleverantör har tillgång till kundens dokument vid vanlig användning. Komponenterna är versionslåsta, OCR-komponenterna kontrolleras mot SHA-256 och språkmodellen är låst till en bestämd revision. SBOM i CycloneDX 1.5 för 0.9.0-beta22, framtagen ur det byggda paketet i bygget, se SBOM.	Bedöma Itha AB enligt artikel 21.3.	SBOM:en har kända luckor och skannas inte mot någon sårbarhetsdatabas. Python-paketen är inte låsta med hashar. Automatisk bevakning av sårbarheter finns bara för npm-beroendena (Dependabot-larm sedan 2026-10-23). Med leverantörerna har Itha AB deras standardvillkor och kan inte visa att kundens krav gäller längre bak i kedjan. Att personuppgiftsbiträdesavtal gäller för one.com, Vercel och Neon är inte bekräftat. Lagring inom EU är inte garanterad hos Cloudflare.
e) Säkerhet vid anskaffning, utveckling och underhåll, inklusive sårbarhetshantering	Utvecklingen beskrivs i Säker utveckling och sårbarhetshanteringen, med fastställda tidsramar för åtgärd, i Sårbarhetshantering. Installer och programfiler signeras av Itha AB. Uppdateringar kontrolleras mot SHA-512, Itha AB:s hela utgivarnamn och den version som manifestet angav, och avvisas när kontrollen inte kan köras (från 0.9.0-beta22). Organisationen kan styra uppdateringarna centralt, se IT-drift.	Styra vilka versioner som körs, och rulla ut säkerhetsrättningar själv om automatiska uppdateringar är avstängda.	Ingen oberoende kodgranskning och inget penetrationstest. Offerter på ett externt penetrationstest av skrivbordsappen och licensservern tas in under hösten 2026; målet är ett genomfört test under Q1 2027. CI spärrar inte release. Uppdateringsmanifestet är inte signerat.

Art. 21.2	Privera och Itha AB	Kunden	Begränsningar i dag
f) Utvärdering av åtgärdernas effektivitet	Kontroller som stoppar bygget, regressionstest för varje bekräftat exportfel och mätning av detektionens träffsäkerhet mot en låst baslinje, som körs för hand. Utkastet till säkerhetsbilaga ger kunden rätt till säkerhetsrevision, dokumentgranskning och intervju.	Egen uppföljning och revision.	Ingen extern granskning och ingen certifiering (ISO 27001, SOC 2). Det finns ingen tredjepartsgranskning att lämna ut.
g) Cyberhygien och utbildning	Programmet visar förslagen och låter användaren bestämma vad som maskeras. Säkerhetsunderlaget beskriver hur en exporterad fil kan kontrolleras.	Utbilda handläggare, särskilt i att granska dokumentet och den sparade filen före utlämnande.	Den automatiska detektionen hittar inte allt. Användaren måste granska filen.
h) Kryptografi	TLS 1.2 eller 1.3 mot licensservern. Sparade granskningar och licensfilen krypteras med AES-256-GCM under en nyckel som skyddas av Windows DPAPI. Licensnycklar lagras som SHA-256-summa på servern. Licensbevis signeras med Ed25519. licens.privera.se skickar HSTS. Se Kryptering och åtkomstkontroll.	Krypterad volym, till exempel BitLocker, för systemdisken och för mappar med original, exporter och loggar.	Exporter, exportloggar, säkerhetsloggen och motorns tillfälliga arbetsfiler, bland annat omaskerade kopior av dokumentet, skrivs okrypterade. Ingen inställning för krypteringsnivå efter informationsklass. Licensnyckeln och kassans session-id står i adressen vid nedladdningen av installern.
i) Personalsäkerhet, åtkomstkontroll och tillgångsförvaltning	Programmet har ingen egen inloggning. Åtkomsten följer Windows-kontot och därmed kundens AD- eller Entra-styrning. Exportloggen och säkerhetsloggen anger vilket Windows-konto som gjorde vad. Installation, uppdateringar och loggning kan styras med grupprincip, Intune och ADMX-mall.	Personliga Windows-konton, eftersom delade konton inte ger spårbarhet till person. Styra vem som får installera och köra programmet, till exempel med AD-grupper, AppLocker eller Intune.	Inga roller i programmet. Förvalt installeras programmet per användare utan administratörsrätt, så kunden behöver själv spärra egen installation. En utgivarregel för Itha AB i AppLocker eller WDAC släpper också in 0.9.0-beta21 och äldre, där den signerade programfilen kan köra annan kod. AppLocker behöver därför en nekande regel för de versionerna, se IT-drift avsnitt 4. Utrullningen är inte provad i Intune eller Configuration Manager. MSI finns inte.
j) Flerfaktorsautentisering och säkra kommunikationer	Programmet har inga egna konton, så kundens MFA vid Windows-inloggningen är det som skyddar programmet. Kommunikationen med licensservern går över TLS.	MFA vid inloggning i Windows eller Entra.	Ingen MFA i programmet. Webbens kundsida och nedladdningen öppnas med licensnyckeln som enda faktor. Vilka av Itha AB:s egna administrativa konton som har MFA är inte dokumenterat.

4. Artikel 20: ledningens ansvar

Ansvaret för cybersäkerheten i Privera ligger hos Itha AB:s ledning. Itha AB har en säkerhetsansvarig, som också är incidentansvarig enligt Incidentrutin. Någon ersättare är inte utsedd.

Kundens ledning ansvarar för kundens åtgärder enligt cybersäkerhetslagen och för att bedöma leverantören. Itha AB lämnar på begäran det underlag som kundens ledning behöver: det här säkerhetsunderlaget, SBOM:en, utkastet till säkerhetsbilaga och svar på frågor.

5. Artikel 23: kundens rapportering av incidenter

En kund som omfattas av lagen ska lämna en tidig varning inom 24 timmar från att den fått kännedom om en betydande incident, en incidentanmälan inom 72 timmar och en slutrapport senast en månad efter incidentanmälan. Så här stöder Itha AB det:

- **Underrättelse till kunden.** Vid en aktivt utnyttjad sårbarhet eller en allvarlig incident som kan påverka kunden underrättar Itha AB berörda kunder utan onödigt dröjsmål, i steg som passar kundens rapportering: tidig varning, uppdatering och slutrapport. Den tidiga varningen ges senast 24 timmar efter att Itha AB fått kännedom, räknat under helgfria vardagar, och vid kännedom på en lördag, söndag, helgdag, midsommarafton, julafton eller nyårsafton senast klockan 12 närmast följande helgfria vardag. Uppdateringen ges senast 72 timmar efter kännedom, räknat på samma sätt. Se Incidentrutin avsnitt 5.
- **Följden för kunden.** När kännedomen uppkommer nära eller under en helg kan Itha AB:s tidiga varning alltså komma senare än kundens egen frist på 24 timmar. Fristerna är satta efter vad en person utan jour kan hålla.
- **Underlag.** Itha AB lämnar det underlag kunden behöver för sin egen anmälan: vad som har hänt, berörda versioner, vidtagna åtgärder och vad kunden kan göra.
- **Avtal.** Fristerna blir avtalsvillkor först i säkerhetsbilagan till avtalet. Den finns som utkast för juristgranskning, och ingen kund har undertecknat den per 2026-10-23. Köpvillkoren nämner inte incidenter.
- **Kvarvarande uppgifter i en fil.** Om kunden hittar en uppgift som skulle ha maskerats i en exporterad fil är det kunden som bedömer om det är en incident eller personuppgiftsincident hos kunden. Itha AB hjälper till att fastställa orsaken och meddelar om felet kan finnas i andra filer.
- **Itha AB:s egen rapportering.** Itha AB rapporterar aktivt utnyttjade sårbarheter och allvarliga incidenter enligt CRA artikel 14, med tidig varning inom 24 timmar. Den fristen gäller alla dagar, men med en person och utan jour kan Itha AB inte garantera att en händelse som inträffar på en helg eller helgdag blir känd inom den tiden.

6. Kontaktvägar

Ärende	Kontakt
Säkerhet, sårbarheter och incidenter	feedback@privera.se, bevakas helgfria vardagar
Dataskyddsfrågor och begäran om underlag	info@privera.se
Support	support@privera.se, helgfria vardagar klockan 09 till 18
Telefon	076 926 35 70 (besvaras i mån av möjlighet)

7. Det som saknas i dag, samlat

- CE-märkning och EU-försäkran enligt CRA. Kravet gäller från 2027-12-11, se Plan för CRA.
- Undertecknad säkerhetsbilaga om incidenter, revision och insyn. Utkast för juristgranskning finns.
- Ersättare för säkerhetsärenden, bevakning på helger och övning av incidentrutinen.
- Penetrationstest och oberoende kodgranskning. Offerter på penetrationstest tas in under hösten 2026, med mål att testet är genomfört under Q1 2027.
- MFA och roller i programmet, och krypteringsnivå efter informationsklass.
- Manipulationsskydd som hindrar ändringar, inte bara upptäcker dem, och insamling provad mot SIEM. Lagring av licensserverns säkerhetslogg.
- Sårbarhetsskanning mot SBOM:en, och automatisk bevakning av motorns Python-paket och OCR-motorns bibliotek.
- Signerat uppdateringsmanifest.
- Garanterad lagring inom EU hos Cloudflare, och bekräftade personuppgiftsbiträdesavtal med one.com, Vercel och Neon.
- MSI-paket och centralt förinstallerad språkmodell.
- Prov av avvecklingsläget i ett installerat program på en ren dator, och avvecklingsnyckeln i förvaring offline.