Säkra leveranskedjor för mjukvara

Arbetsrubrik: ”Aspekter för hantering av leveranskedjans säkerhet för programvara med direkta och transitiva beroenden på mjukvarukomponenter”

Kevin Karlsson

2023

Referat

Nyckelord: mjukvarans leveranskedja, beroendehantering, säkerhet, sårbarheter

Keywords: software supply chain, dependency management, security, vulnerability

Innehållsförteckning

1. Introduktion

Säkerhet har en alltjämt växande roll inom mjukvarubranschen. Under de senaste åren har vi upprepade gånger fått bevittna hur även världens största mjukvarugiganter har fått bekämpa säkerhetsproblem och sårbarheter i sina produkter.

Säkerhetsproblem i produkter kan direkt uppstå på grund av utomstående aktörers handlingar, men det är vanligare att mjukvaran är sårbar på grund av oväntat eller felaktigt beteende, och att en aktör utnyttjar sådana svagheter för sina egna syften.

Alltid är sårbarheterna inte orsakade direkt av företagens egen mjukvara, utan ibland kan det vara på grund av sårbarheter i mjukvarubibliotek från externa parter som inkluderats i programvaran. Världsomspännande kalabalik uppstår då sårbarheter uppdagas i populära mjukvarubibliotek som vanligen inkluderas i programvara.

Programvara som en produkt för slutanvändaren; privatperson eller företagsanvändare, består ofta av sammankopplade mjukvarukomponenter, där varje enskild del fyller en viss funktion och bidrar med funktionalitet till helheten. Mjukvarukomponenter kan utgöra mjukvarubibliotek med gemensam funktionalitet som flera olika produkter förlitar sig.

Mjukvarubibliotek tillsammans med kod sammanställs till programvara som levereras till kunden. Detta utgör mjukvarans leveranskedja. Syftet med denna avhandling är att gå igenom säkerhetsaspekterna för denna leveranskedja. Speciell uppmärksamhet ges hanteringen av mjukvarubibliotek, och åtgärder som kan vidtas vid upptäckt av säkerhetsproblem eller sårbarheter i mjukvarubibliotek med syfte att åtgärda bristerna.

Avhandlingen reflekterar situationen utifrån en förenklad synvinkel ämnad för mjukvaruföretag som levererar mjukvara. {I fall av cloud solutions, SaaS etc måste man tillämpa lite…}

2. Bakgrund

a. Leveranskedjan

Leveranskedjan för mjukvara utgör de steg som mjukvara genomgår för att bilda en produkt som levereras till kunden. Produkt såsom det avses här kan till exempel vara mjukvarusystem, mjukvarubibliotek, programvara eller uppdateringar till existerande mjukvarusystem. {I produkten som levereras kan utöver mjukvaran även ingå instruktioner och bruksanvisningar osv.}

Som vi kommer att inse nedan kan leveranskedjor kedjas, så att produkter såsom mjukvarubibliotek utgör delar i leveranskedjan för någon annan mjukvara.

<<INSERT PICTURE HERE>>

Nedan gås de olika delarna av leveranskedjan igenom i logisk ordning.

Kod

Källkod utgör grunden för mjukvara. Inom ramen för denna avhandling avser kod det som mjukvaru-utvecklare inom företag producerar som en del av mjukvaran. Källkod lagras i kodförråd som används för versionshantering, mera om detta nedan.

{Öppen källkod. Licensiering. Proprietär. ??}

Mjukvarubibliotek

I stället för att uppfinna hjulet på nytt för varje mjukvara är det inom mjukvaru-utveckling möjligt att utnyttja existerande kod i form av mjukvarubibliotek för att inkludera funktionalitet i mjukvara som produceras. Bibliotek som mjukvara beror på utgör mjukvaruberoenden (dependencies). Programmeringsspråk erbjuder olika metoder och tekniker för at hantera mjukvarubibliotek och -beroenden.

{Beroendehantering, transitiva beroenden, versionering…}

Versionshanteringssystem

Källkod lagras i ett kodförråd, versionshanteringssystemet, som gör det möjligt att följa med ändringar i koden och hantera olika versioner av koden.

{Vad är en version? Kodversion vs programversion.}

Byggsystem

Ett byggsystem används för att bygga mjukvara. Systemet kombinerar källkod och mjukvaruberoenden enligt givna parametrar och inställningar till en mjukvara.

Mjukvarukatalog

Mjukvarubibliotek samt slutliga produkter lagras och hanteras av mjukvarukataloger. Byggsystem hämtar mjukvarubibliotek ur kataloger, men kan även lagra mjukvara i den.

Paketering / Pipelines

Mjukvaran hämtas ur mjukvarukatalogen och färdigställs tillsammans med resten av leveransen. Detta kan ske automatiskt eller manuellt. {Dokumentation, installer etc jfr (build) pipelines etc.}

Leverans

Den färdiga produkten levereras till kunden; fysiskt, digitalt eller imaginärt. {Leverans av SaaS??}

Mjukvaruberoenden

{Mjukvara komponeras oftast {SOURCE} av mjukvarubibliotek samt mjukvarans egen källkod.}

Transitiva beroenden

3. Aspekter

I det följande kapitlet kommer vi att gå igenom riskerna i de olika delarna av leveranskedjan.

Inte starkare än den svagaste länken.

Trust. Transitive trust? Who do you trust? Något exempel? Trust trust och mera trust.

Attacker mot leveranskedjor ter sig speciellt lockande, eftersom illvilliga aktörer potentiellt kan komma åt att påverka kundernas kunder bara genom att hacka denna ena svaga länk.

Populära attacker? Eftersom man då når ut till kundernas kunder med endast ett hack.

{Är det out of scope ifall man hackas och källkoden blir stulen…}

Påverkan/effekten av sårbarhetsaspekterna. Hur kritiska är de? Varför bryr man sig?

Säkerhetsaspekter av mjukvarukomponenter (4-6 sidor)

Attacker och sårbarheter

4. Hantering

I det följande kapitlet kommer vi att gå igenom hur man säkrar leveranskedjan, med speciell fokus på (automatisk beroendehantering).

Säkra sina system.

Egen mjukvarukatalog. Review FOS?

Eller? Processen för att hantera en CVE…

Hur ta säkerhetsaspekterna i beaktande i leveranskedjan? (4-6 sidor)

Processen som efterföljer att en sårbarhet har identifierats

Förklara olika delar av processen i större detalj

5. Sammanfattning

Sammanfattning (1-2 sidor)

Förklara problematiken

Avhandlingens kontribution

Framtida visioner

6. Referenser

There are no sources in the current document.