



UPPSALA
UNIVERSITET

Dnr UFV 2014/1307

Säker systemutveckling

Rutiner för Informationssäkerhet

Fastställda av Säkerhetschef
Senast rev.

2014-10-28
2021-04-29

Innehållsförteckning

1	Inledning	3
2	Ansvar	3
2.1	Efterlevnad	3
2.2	Uppdatering av riktlinjerna	3
3	Definitioner	4
4	Omfattning	4
4.1	Övergripande	4
4.2	Identifiering av säkerhetskrav före utveckling	5
4.3	Källkod	6
4.4	Kodsignering	6
4.5	Säker kommunikation/säker lagring av information	6
4.6	Autentisering/åtkomst till databaser, filer och källkod	7
4.7	Loggning	7
4.8	Test och testdata	7
4.9	Överlämnande till drift/fortsatt förvaltning av systemet	7

1 Inledning

Dagens internetbaserade IT-miljöer, ständigt utsatta för nya hot relaterade till säkerhet, ställer särskilda krav i allt arbete som bedrivs med utveckling och underhåll av IT-baserade system. Att använda utvecklingsmetoder och rutiner som inkluderar moment där särskilt fokus ligger på säkerhetsmässiga aspekter utgör en grundläggande förutsättning för säker drift av de system som utvecklas.

Då arbete med systemutveckling kräver ständig bevakning av risker inom området och hantering av dessa rekommenderar vi alla systemutvecklare att kontinuerligt bevaka information från den globala organisationen Open Web Application Security Project (OWASP), <https://www.owasp.org>. För råd och stöd, kontakta säkerhetsavdelningen.

Rutinerna baseras på *rutiner för informationssäkerhet* (UFV 2017/93).

2 Ansvar

2.1 Efterlevnad

Säkerhetschefen ansvarar för att rutinerna kontinuerligt uppdateras och att underliggande stöddokument fastställs.

Ansvar för efterlevnad av rutinerna fördelar sig enligt följande:

- *Projektägare/motsvarande* har det övergripande ansvaret i samband med systemutvecklingsprojekt.
- *Systemägare, e-områdesansvarig/motsvarande* har det övergripande ansvaret i det löpande förvaltningsarbetet.
- *Projektledare, e-koordinator/motsvarande* ansvarar för att rutinerna beaktas i det dagliga arbetet med systemutveckling samt att eventuella konsulter har kännedom om dessa.
- *Systemutvecklare* ansvarar för att rutiner med direkt påverkan på utvecklingsarbetet följs.

2.2 Uppdatering av riktlinjerna

Säkerhetschefen ansvarar för att riktlinjerna kontinuerligt uppdateras och att underliggande stöddokument fastställs.

3 Definitioner

Informationsklassificering utgör ett moment där information som hanteras, exempelvis av IT-system, bedöms utifrån aspekterna konfidentialitet (sekretess), riktighet och tillgänglighet. Informationens behov av säkerhetsmässiga åtgärder bestäms i en behovsskala bestående av nivåerna basnivå, hög nivå samt särskilda krav.

Exempel på säkerhetsproblem att beakta vid systemutveckling:

- *SQL injection* är ett sätt att utnyttja säkerhetsproblem i hanteringen av indata i system som arbetar mot en databas. Injektionen sker genom att en användare skickar in parametrar till en databasfråga, utan att parametrarna transformeras korrekt med avseende på speciella tecken. Med anpassade parametrar kan en användare kringgå inloggningssystem och manipulera data.
- *Cross Site Scripting, XSS*, används som metod för att stjäla information eller förstöra en webbsidas utseende. Metoden är tillsammans med SQL injection omnämnd som en av de mest kritiska säkerhetsriskerna med koppling till webbapplikationer.

Multifaktorautenticering eller *flerfaktorsautenticering* – är en metod för att bekräfta en användares identitet med två eller flera olika faktorer, vilket ger en högre säkerhet än enbart ett lösenord. Faktorerna består av något man har (kort, dosa, mobiltelefon (Mobilt BankID)), något man vet (lösenord, PINkod) och något man är (biometri, t.ex. fingeravtryck). Tvåfaktorsautenticering (2FA) kallas det fall där enbart två faktorer används.

Avidentifiering av data innebär att alla möjligheter att koppla information till en unik person är borttagna.

4 Omfattning

4.1 Kritiska säkerhetsrisker

Exempel på säkerhetsproblem att beakta vid systemutveckling:

- *SQL injection* är ett sätt att utnyttja säkerhetsproblem i hanteringen av indata i system som arbetar mot en databas. Injektionen sker genom att en användare skickar in parametrar till en databasfråga, utan att parametrarna transformeras korrekt med avseende på speciella tecken. Med anpassade parametrar kan en användare kringgå inloggningssystem och manipulera data.
- *Cross Site Scripting, XSS*, används som metod för att stjäla information eller förstöra en webbsidas utseende. Metoden är tillsammans med SQL injection omnämnd som en av de mest kritiska säkerhetsriskerna med koppling till webbapplikationer.

4.2 Skydd av information

Skydd av känsliga data

Universitetets IT-system ska vara utformade i enlighet med kraven i offentlighetslagstiftningen och andra tillämpliga lagar och regler. Om osäkerhet råder ska universitetets juridiska avdelning tillfrågas.

Information som hanteras i universitetets IT-system ska vara omgärdad av ett skydd som motsvarar känsligheten i denna. Detta ska säkerställas på ett tidigt stadium i utvecklingsprocessen - se nedan under 4.3, *Identifiering av säkerhetskrav före utveckling*.

Skydd av personuppgifter

De grundläggande principerna för inbyggt integritetsskydd (*privacy by design*) ska vara vägledande:

- System ska vara designade för att samla in så få uppgifter som möjligt - enbart sådana personuppgifter som är nödvändiga för syftet
- System ska behålla uppgifter under så kort tid som möjligt - uppgifter ska raderas/anonymiseras så fort behovet av identifiering upphört
- Så få användare som möjligt ska ha tillgång till personuppgifter - enbart användare som har en yrkesmässig anledning
- Uppgifter ska kunna skyddas under hela sin livscykel - insamling, transport, säkerhetskopiering, utrantering och destruering av lagringsmedia m.m.

Skydd mot intrång och driftstörningar genom yttre påverkan

Metoder för att säkerställa korrekt bearbetning ska användas. Dessa metoder bör innefatta moment som kodgranskning, rutiner för validering av indata samt rimlighetskontroller av utdata. Det är av största vikt att den programmeringsmetodik som används hålls aktuell, detta både med avseende på användning av nya tekniker och skydd mot säkerhetsmässiga hot.

Kontakta Säkerhetsavdelningen för råd och stöd.

4.2 Identifiering av säkerhetskrav före utveckling

Säkerhetskrav ska identifieras och dokumenteras i ett utvecklingsprojekts inledande fas. Systemägare och projektledning ansvarar gemensamt för att momenten informationsklassificering, kravanalys och riskbedömning genomförs i enlighet med *rutiner för riskhantering* (UFV 2018/211). Kontakta säkerhetsavdelningen för råd och stöd.

Planering och uppföljning av skyddsåtgärder ska göras som en del av projektarbetet, så länge projektet pågår. Efter överlämning från projekt till förvaltning ska motsvarande ingå som löpande åtgärder i förvaltningsarbetet. Ansvarig för att så sker är systemägare/motsvarande.

4.3 Källkod

Åtkomst till källprogramkod som ägs av universitetet ska kontrolleras av ett behörighetssystem.

Källkod som ägs av universitetet ska versionshanteras. Alla förändringar som görs i koden ska loggas.

Lösenord och krypteringsnycklar får inte finnas i källkoden. Dessa ska lagras i konfigurationsfiler som inte versionshanteras.

4.4 Kodsignering

Kodsignering innebär att man signerar programvara för distribution, för att intyga att certifikatinnehavaren/organisationen står bakom den färdiga produkten.

Eftersom den typen av signering normalt får till följd att produkten/programmet installeras hos mottagaren utan varningar, är det av yttersta vikt att man faktiskt kan garantera att programvaran är fri från skadligt beteende såsom virus, trojaner, etc.

Det innebär i praktiken att signaturen är en form av kvalitetsstämpel från organisationen, och certifikatet som används måste därför hanteras med stor försiktighet. Den privata nyckel som hör till certifikatet får inte kopieras och förvaras på flera ställen med mindre än att en process finns på plats för att säkerställa att onödiga kopior raderas och behovsprövning sker vid varje tillfälle.

Om signeringsnyckeln måste inkluderas i automatiska bygg-, test- eller distributionsprocesser, omfattas dessa processer av samma säkerhetskrav som signeringsnyckeln; det måste finnas tillräckligt stark autentisering och auktorisering kopplad till dessa processer för att kunna säkerställa att ingen obehörig programkod, eller programkod från obehörig person, kan fås signerad, vare sig av misstag eller med vilja.

Interna "test"-nycklar ska, om möjligt, användas för kodsignering under utvecklings- och testfasen.

Själva slutsigneringen med skarp signeringsnyckel ska ske manuellt under noggrant styrda förutsättningar, gärna med krav på att två eller flera personer bidrar med varsin del av nyckeln.

4.5 Säker kommunikation/säker lagring av information

Dataöverföringar av känsliga data ska alltid ske via krypterad transport, HTTPs-anslutning eller motsvarande.

Vid överföring av känslig information till annat system ska motsvarande skyddsåtgärder som vidtas inom det egna systemet vara vidtagna i kommunikationen samt i det mottagande systemet.

4.6 Autentisering/åtkomst till databaser, filer och källkod

Universitetets standardiserade användaridentiteter och autentiseringssystem ska användas så långt det är möjligt.

Systemanvändare ska ha en unik användaridentitet, dvs. inga gruppidentiteter får finnas.

Lösenord ska följa universitetets *rutiner för lösenordshantering* (UFV 2013/1490).

Multifaktorautentisering ska användas för åtkomst till IT-system eller tjänster som enligt rutinerna för informationsklassning innehåller konfidentiell information.

Åtkomst till databaser, filer och källkod till programmen ska styras på ett säkert sätt. Dokumenterade rutiner för behörighetstilldelning ska finnas och följas. Det ska gå att begränsa användares åtkomst till system och databaser utifrån parametrar som roll, organisationstillhörighet och liknande. Se *rutiner för Hantering av behörigheter och roller* (UFV 2018/1170).

4.7 Loggning

Systemet ska inkludera rutiner som skapar erforderliga loggar för uppföljning av säkerheten i systemet.

Universitets rutiner för loggning ska följas. Dessa beskrivs i *rutiner för anskaffning och drift av IT-system* (UFV 2020/2599).

4.8 Test och testdata

Utvecklings- och testarbete ska utföras i en egen, från driften väl avskild, miljö.

I ett fall då data i en testmiljö är baserad på verkligt material, ska testmiljön omgärdas av säkerhetsåtgärder motsvarande de som används i produktionsmiljön.

Om det finns alternativ till att använda verklig produktionsdata i testmiljön, t.ex. att använda fingerade eller avidentifierade uppgifter, ska dessa användas istället. Detta gäller även om det blir mer kostsamt.

Rutiner ska finnas för att säkerställa att testdata kontrolleras och skyddas.

4.9 Överlämnande till drift/fortsatt förvaltning av systemet

Vid driftsättning ska systemet överlämnas till systemförvaltning enligt en dokumenterad rutin för detta. En fastställd organisation för systemförvaltning ska finnas och vara känd av både överlämnande och mottagande parter.

Dokumentation som beskriver systemet och dess kopplingar till andra system ska finnas framtagen innan systemet överlämnas för drift och fortsatt förvaltning.

Rutiner för godkännande av systemförändringar ska finnas och följas.

Alla föreslagna systemändringar ska granskas för att säkerställa att de inte äventyrar säkerheten vare sig i systemet eller i driftmiljön.