MalaRIA proxy – en av farene ved usikker konfigurasjon

Erlend Oftedal, BEKK

Communities in action

10. mai 2010

The OWASP Foundation
http://www.owasp.org

Usikker konfigurasjon

- Eksempler
  - Default passord
  - Ubrukte tjenester
  - Ubeskyttede kataloger og tjenester

Attack Vectors	Security Weakness		Technical Impacts
Exploitability EASY	Prevalence COMMON	Detectability EASY	Impact MODERATE

Hvor?

- Kan skje på mange nivåer:
  - OS
  - Webserver
  - Applikasjonsserver
  - Rammeverk

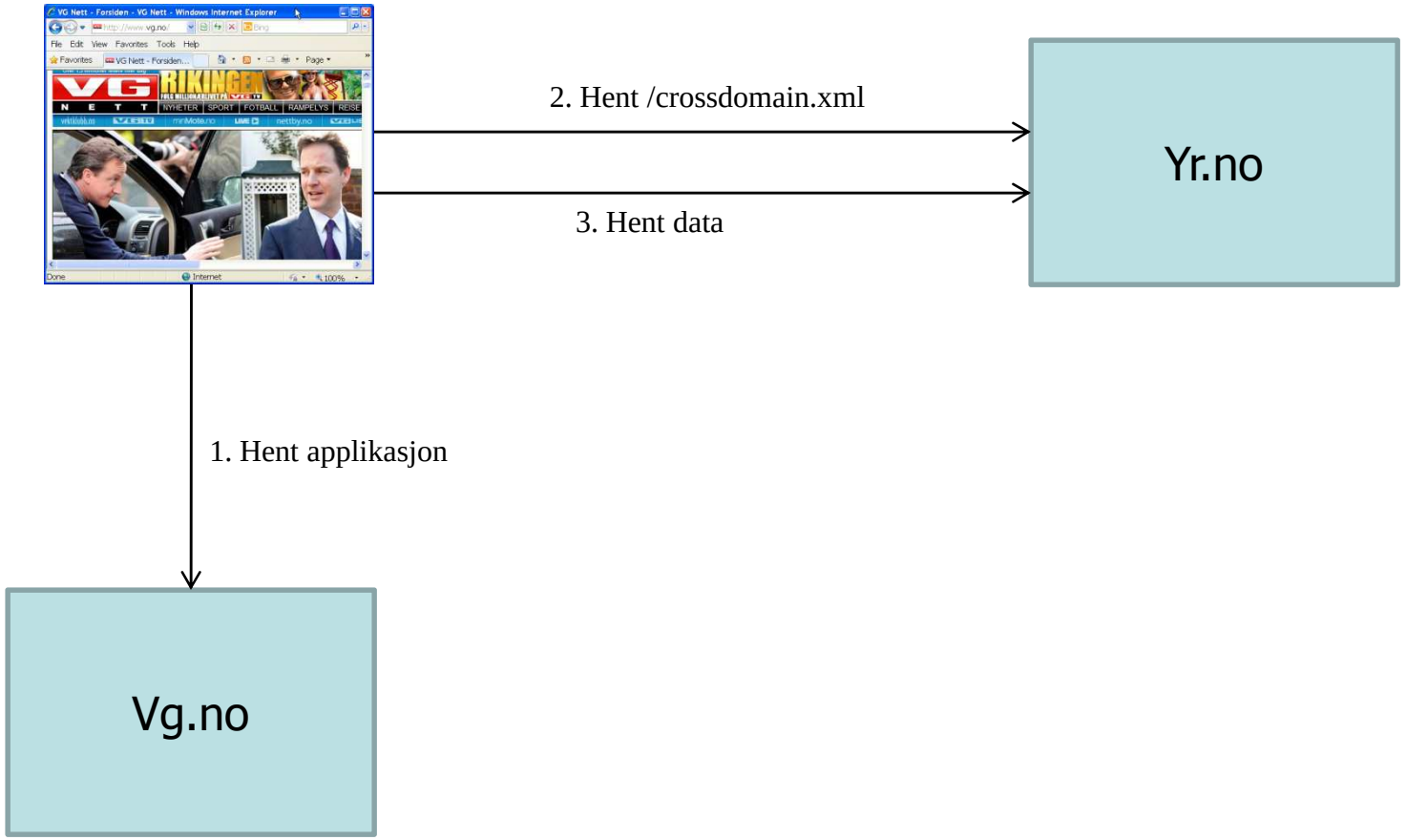
Eksempler

- Applikasjonsserverens admin-interface er tilgjengelig fra internett
- Gjestekontoer eller testkontoer er tilgjengelig i produksjon med default passord
- En REST-tjeneste som tilbyr konfidensielle data og som ikke er beskyttet (kan kalles uten at man er logget inn)

Cross domain policy

- Same-origin policy begrenser hvilke domener en webapplikasjon kan kontakte
- Flex og Silverlight støtter bruk av policyfiler som kan åpne for kommunikasjon med andre domener

Bruk av cross domain policy



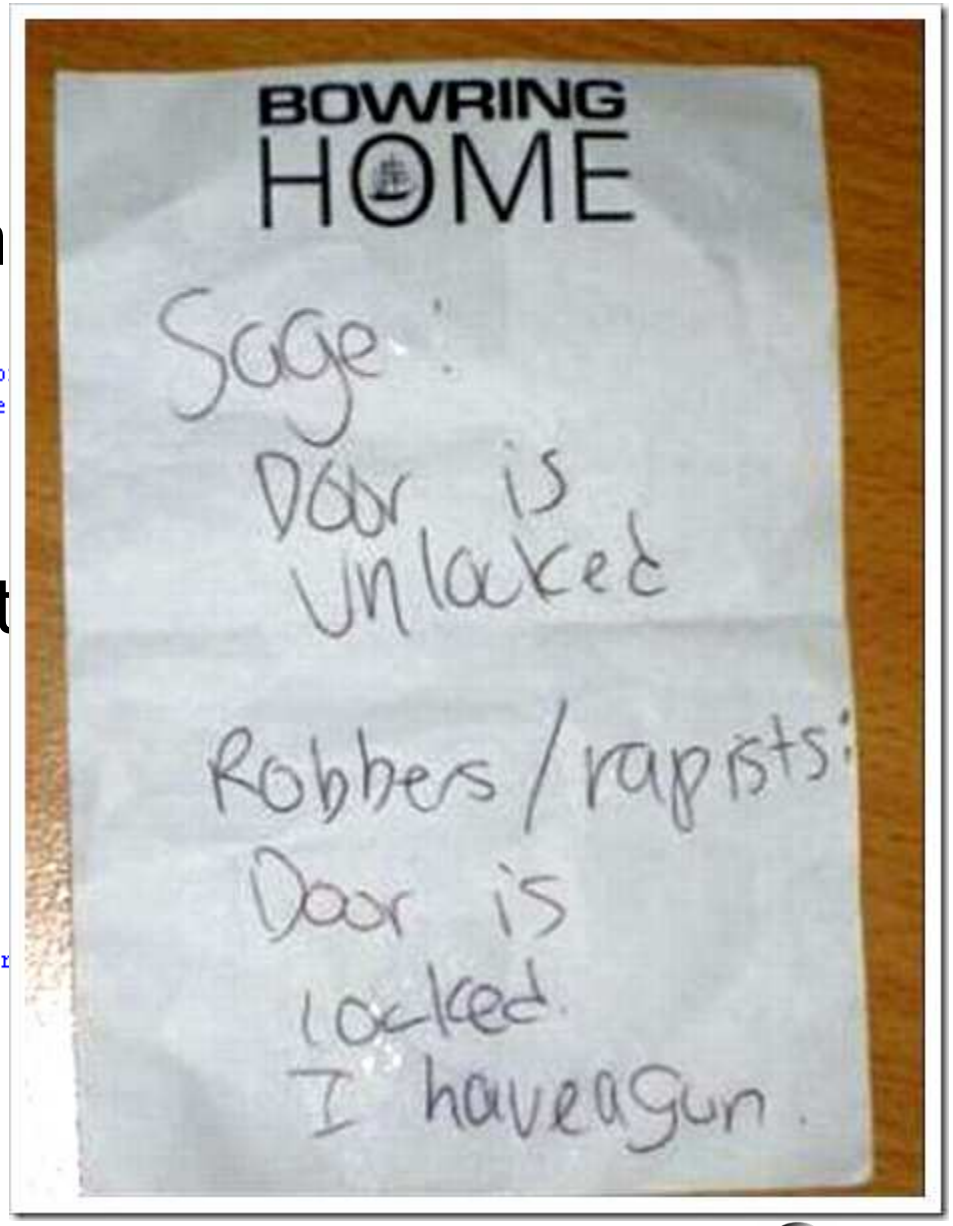
Usikker konfigurasjon

- Flex – åpen crossdoma

```
<?xml version="1.0" encoding="UTF-8"?>
<cross-domain-policy xmlns:xsi="http://www.w3.o
xsi:noNamespaceSchemaLocation="http://www.adobe
  <allow-access-from domain="*" />
</cross-domain-policy>
```

- Silverlight – åpen client

```
<?xml version="1.0" encoding="utf-8"?>
<access-policy>
  <cross-domain-access>
    <policy>
      <allow-from http-request-headers="*">
        <domain uri="*"/>
      </allow-from>
      <grant-to>
        <resource path="/" include-subpaths="tr
      </grant-to>
    </policy>
  </cross-domain-access>
</access-policy>
```



Åpen policy tillater CSRF mot siten

- Sikkerhetsmiljøet har advart mot dette siden 2006
  - Likevel finnes slike åpne filer ofte i produksjon
- En hvilket som helst server kan tilby sider som forårsaker forespørsler mot siten på vegne av brukeren
- Offerets cookies og lignende sendes med

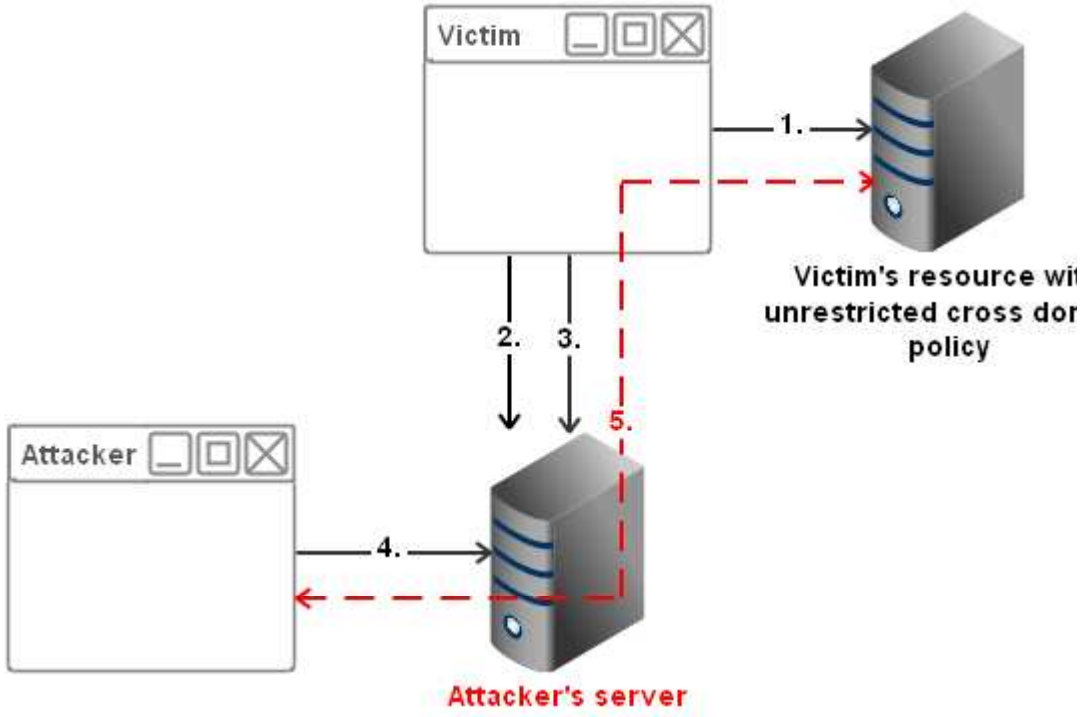
MalaRIA proxy

- Malicious RIA
- Demoapplikasjon for å demonstrere hvor farlig usikre policies er
- Lar en angriper bruke en flex/silverlight applikasjon som en proxy
  - Angriper bruker offerets browser som proxy
- Java backend og støtte for både flex og silverlight frontend

Støtte

- GET/POST
- Nedlasting av
  - HTML
  - Javascript/CSS
  - Binære data (bilder, pdfer osv.)

Oppsett

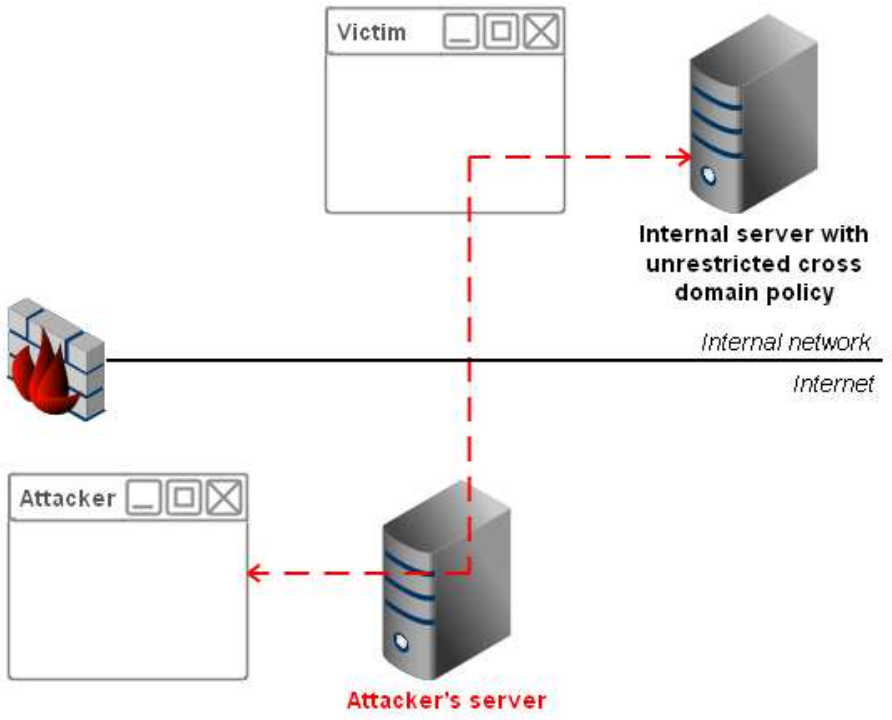


1. Bruker logger inn på siten med en åpen policy
2. Bruker lures til å besøke angripers server
3. Proxyen lastes
4. Angriper kobler til backend
5. Angriper surfer via offerets browser

Demo

Hva med interne servere med åpne policies?

- Kan angripes, så sant angriper vet url



Hva gjør jeg for å sikre meg?

- Begrens hvilke servere som kan koble seg tilbake til din site

```
<cross-domain-policy>
   <allow-access-from domain="*.vg.no"/>
   <allow-access-from domain="*.emediate.se"/>
   <allow-access-from domain="*.fvn.no"/>
   <allow-access-from domain="*.aftenbladet.no"/>
   <allow-access-from domain="*.aftenposten.no"/>
   <allow-access-from domain="*.finn.no"/>
   <allow-access-from domain="*.bt.no"/>
   <allow-access-from domain="*.e24.no"/>
   <allow-access-from domain="*.meglergaarden.no"/>
   <allow-access-from domain="heliosiq.adtech.de"/>
   <allow-access-from domain="adtech.panthercustomer.com"/>
   <allow-access-from domain="aka-cdn-ns.adtech.de"/>
   <allow-access-from domain="aka-cdn.adtech.de"/>
   <allow-access-from domain="annonse.kroma.no"/>
</cross-domain-policy>
```

Men jeg har bare åpne data

- Hvis man ikke har pålogging og kun tilbyr åpne data, kan man ha en åpen policy
  - Men det anbefales ikke da man fort kan overse noe
  - Husk at for eksempel påmelding til nyhetsbrev, kontaktskjema osv. er potensielle angrepspunkter

Men jeg har både åpne og konfidensielle data

- Man kan ha en åpen policy som kun gjelder for en spesifikk folder (og subfolders)
- Flex
  - Egen policy fil i katalogen som skal være åpen
  - Security.loadPolicyFile(url)
- Silverlight:

```
<?xml version="1.0" encoding="utf-8"?>
<access-policy>
  <cross-domain-access>
    <policy>
      <allow-from http-request-headers="*">
        <domain uri="*"/>
      </allow-from>
      <grant-to>
        <resource path="/" include-subpaths="true"/>
      </grant-to>
    </policy>
  </cross-domain-access>
</access-policy>
```

Spørsmål?

- Mer info:
  http://erlend.oftedal.no/blog/?blogid=107
- Stygg PoC kode:
  http://github.com/eoftedal/MalaRIA-Proxy

Bli med i OWASP Norway og lær mer om sikkerhet i webapplikasjoner. Det er gratis å være med!

http://www.owasp.org/index.php/Norway