



# The Future of XSS

by **Sven Vetsch / Disenchant**

## Sven Vetsch / Disenchant

- Working for Dreamlab Ltd.
  - IT Security Tester
  - IT Security Analyst
  - IT Security Engineer
  
- Main interests
  - Webapplication Security
  - Social Engineering
  
- [www.disenchant.ch](http://www.disenchant.ch)

# Thanks to

- Jeremiah Grossman
  - [www.whitehatsec.com](http://www.whitehatsec.com)
- Petko Petkov (aka. pdp)
  - [www.gnucitizen.org](http://www.gnucitizen.org)
- Johann-Peter Hartmann
  - [www.mayflower.de](http://www.mayflower.de)

# Table of Content

- Why this Topic
- XSS Basics
- Protection
- XSS Today
- Other XSS related Attacks
- “Future” Scenarios

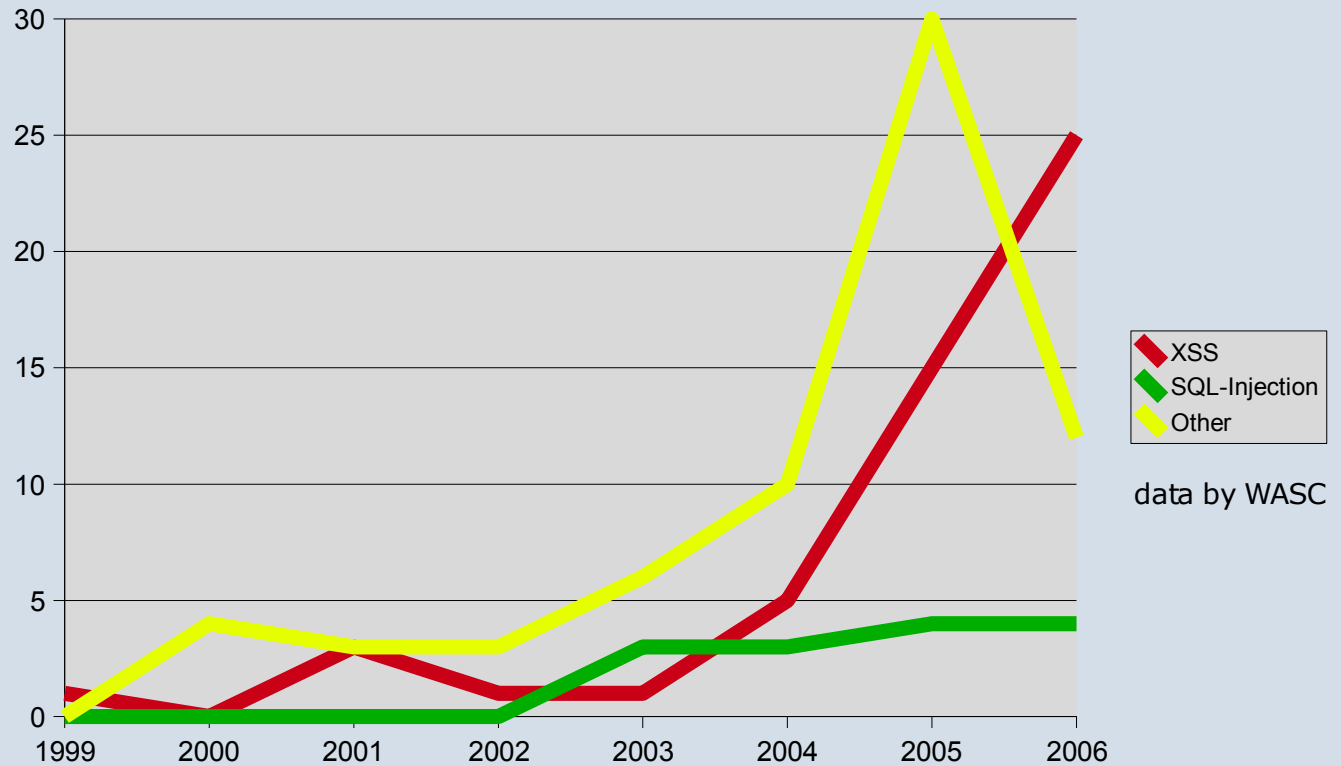


## Why this Topic

## Why this Topic

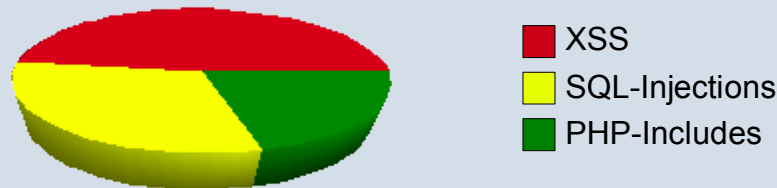
- Actuality
- Enormous potential of XSS
- Most People also “experts” still don't realize that XSS could be very dangerous
- Everyone can find some easy XSS

# Statistics Webapp Security



## Common Vulnerabilities and Exposures (CVE)

- Reported 4375 security issues in the first nine months of 2006
- Web-related flaws have captured the top three spots
  - 21.5 percent cross-site scripting (XSS)
  - 14 percent SQL Injection
  - 9.5 percent php includes

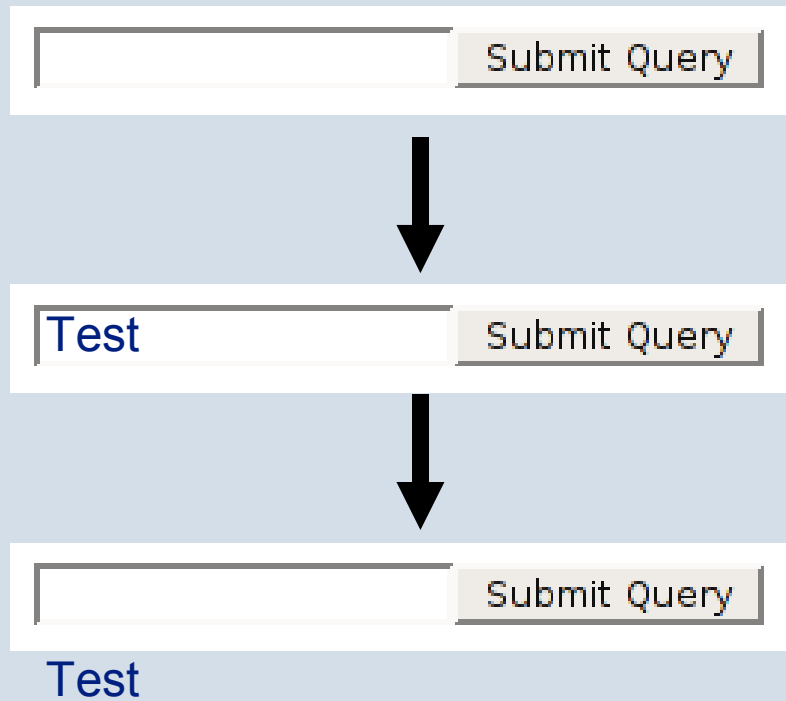


- Buffer overflows came fourth, at 7.9 percent.

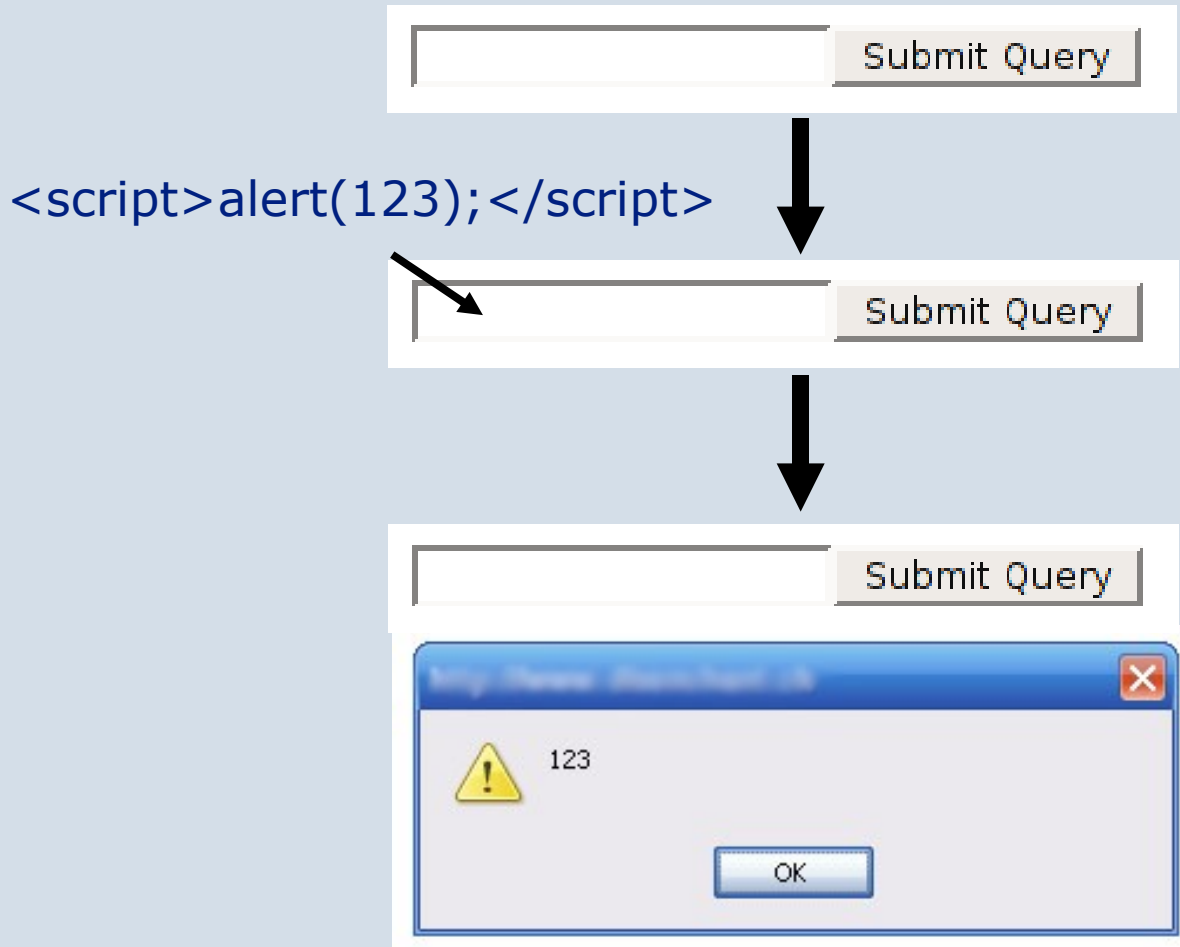


## XSS Basics

## How it works



## How it works



## How it works

```
<html>  
<body>  
  
  <?php  
    echo $string;  
  ?>  
  
  <form method="post">  
    <input name="string">  
  </form>  
  
</body>  
</html>
```

## Types

- temporary
  - This type is only effective, when there's a user interaction. For example clicking a link.
  - The normal XSS published on FD and Bugtraq
- permanent
  - Very dangerous because it doesn't need any interaction of the user.

## The big Problems

- HTTP --> UFBP
  - Universal Firewall Bypass Protocol
- More and more Hardware components have a webinterface
- There are uncountable targets out there
- Increasing knowledge in Javascript



## Protection

## Basic Protection Types

- Patching and Anti-Virus
- Corporate Web Surfing Filters
- Security Socket Layer (SSL)
- Two factor authentication
- Stay away from questionable websites
- Disable Javascript

## White- / Black-Listing

- Used especially if you have to allow some HTML-Tags
- Black-Listing
  - Bad idea !!!
  - You never would have all possible attacks listed
- White-Listing
  - Better then Black-Listing at all
  - It could be that an Attacker can use one of the allowed tags to start an attack
- At the moment, we don't have a universal remedy against XSS



## XSS Today

## XSS Today

- As we've seen it's the most used attack type at the moment
- Most companies which own webapps think that they're not responsible for their customers machine
  - Won't fix their webapps

# What to do with XSS

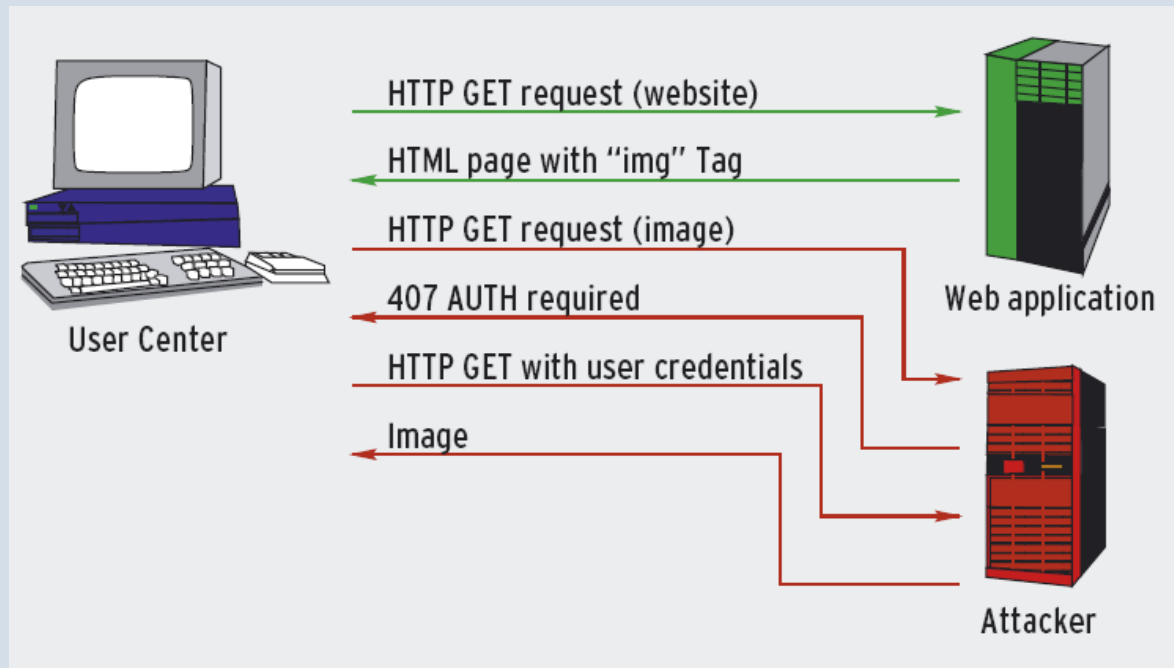
- Cookie- / Session-Hijacking
- Phishing
- Defacements
- ...



## Other XSS related Attacks

only a few examples

# XSA



### „GIF“-Bug

- A Bug in the MS Internet Explorer
- Unpatched (found and reported 15 July 2005)
- Including IE 7 Beta (final not tested yet)
- Not only GIF, nearly every format
- A user must directly request the corrupt file
- Bypassing many image uploads

```
<GIF89a 8 f >  
<html>  
  <head>  
    <script>  
      alert("XSS");  
    </script>  
  </head>  
</html>
```

## Self-contained XSS Attacks

- Only works in Firefox and Opera
- Example
  - `data:text/html;base64,PHNjcmlwdD4KYWxlcuQoIkhpIDBzZWMgOikiKTsKPC9zY3JpcHQ+`
- We can also generate (malicious) binaries this way

## Self-contained XSS Attacks

- How to execute
  - Put it directly in the URL field of the webbrowser and press Enter :P
  - Looking for a redirecting vulnerability in a webapp and put it there.
  - Make a redirecting through a XSS vulnerability.
  - Sending a Mail with such a Link in it to someone.



## „Future“ Scenarios

but it's really nothing new ;)

## XSS-Backdooring

- Put XSS in other file formats
  - Flash-Movies
  - PDFs
  - Quicktime-Movies
  - ...
- Demo with a Flash-Movie

# XSS-Worms

- Platform independent
- Samy
  - He had no friends but he knows how to write something in Javascript
  - Infected MySpace.com
  - After about 24h over 1'000'000 infections/friends
  - MySpace.com had to shut down their servers for cleaning up
- Yamanner
  - Infected Yahoo!-Mail
  - Nobody knows how many mail addresses were stolen for spamming purposes

# XSS-Worms



Profile User A



Profile User B



User A



User B

# XSS-Worms



Profile User A



Profile User B



User A

XSS



User B

# XSS-Worms



Profile User A



Profile User B



User A



User B



XSS

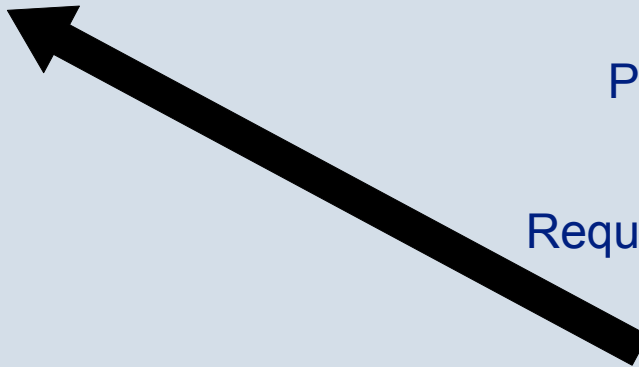
# XSS-Worms



Profile User A



Profile User B



Request



User A



User B

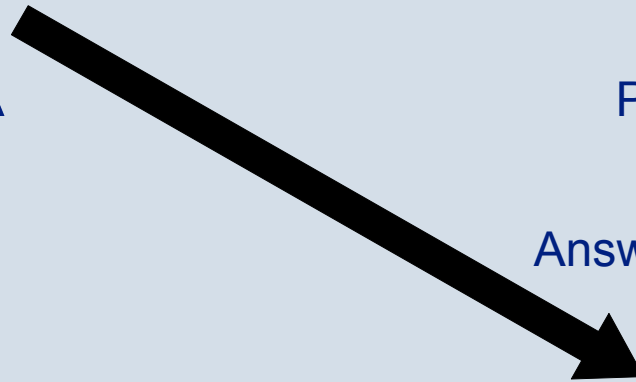
# XSS-Worms



Profile User A



Profile User B



Answer with the XSS

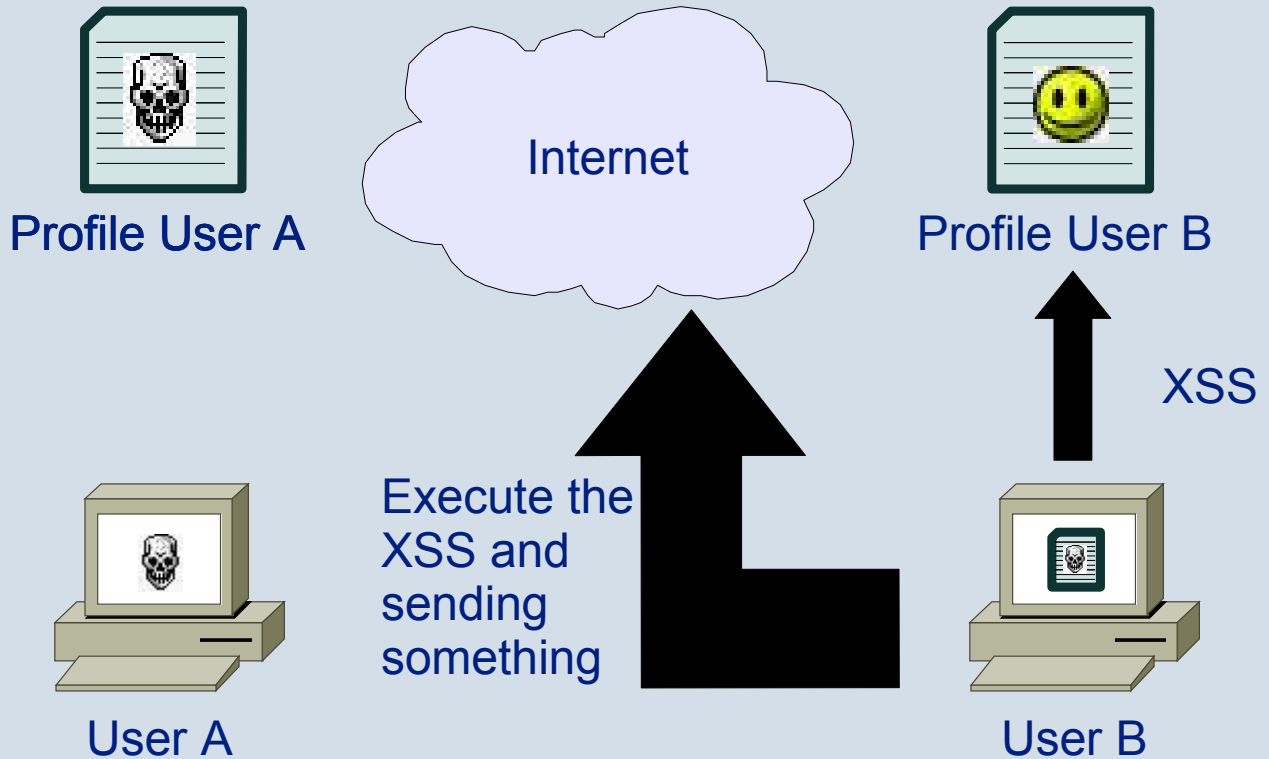


User A



User B

# XSS-Worms



# XSS-Worms



Profile User A



Profile User B



User A

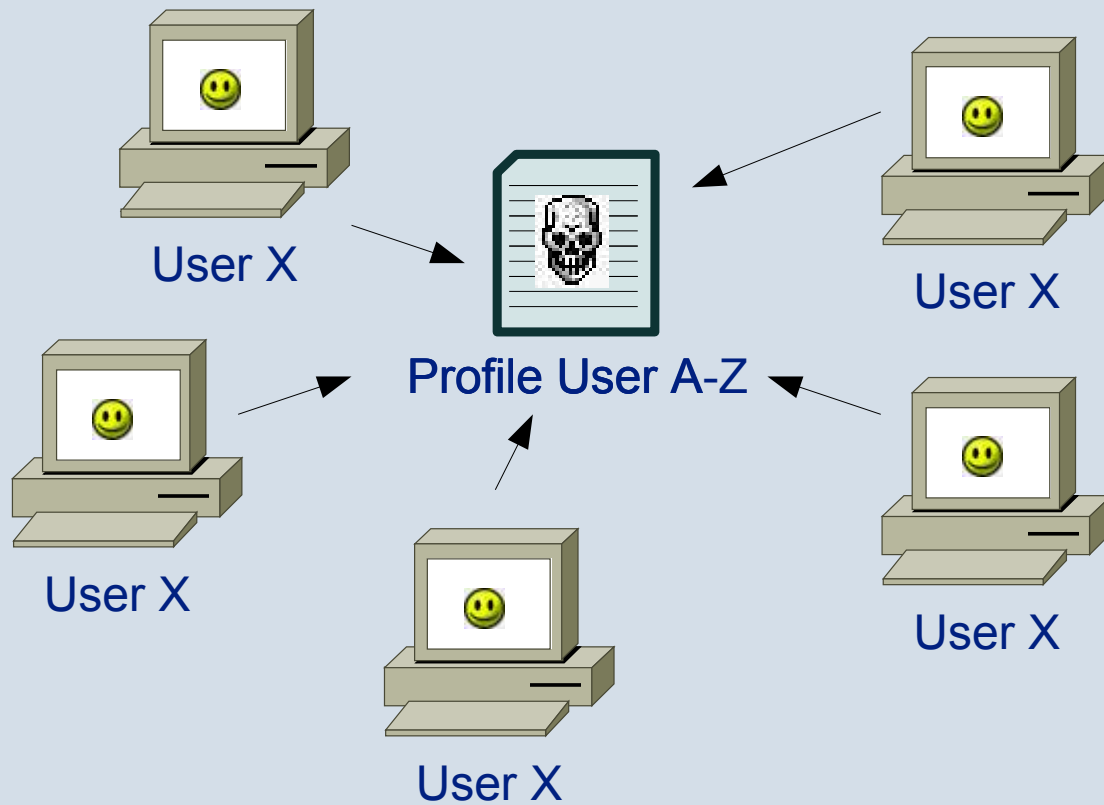


User B

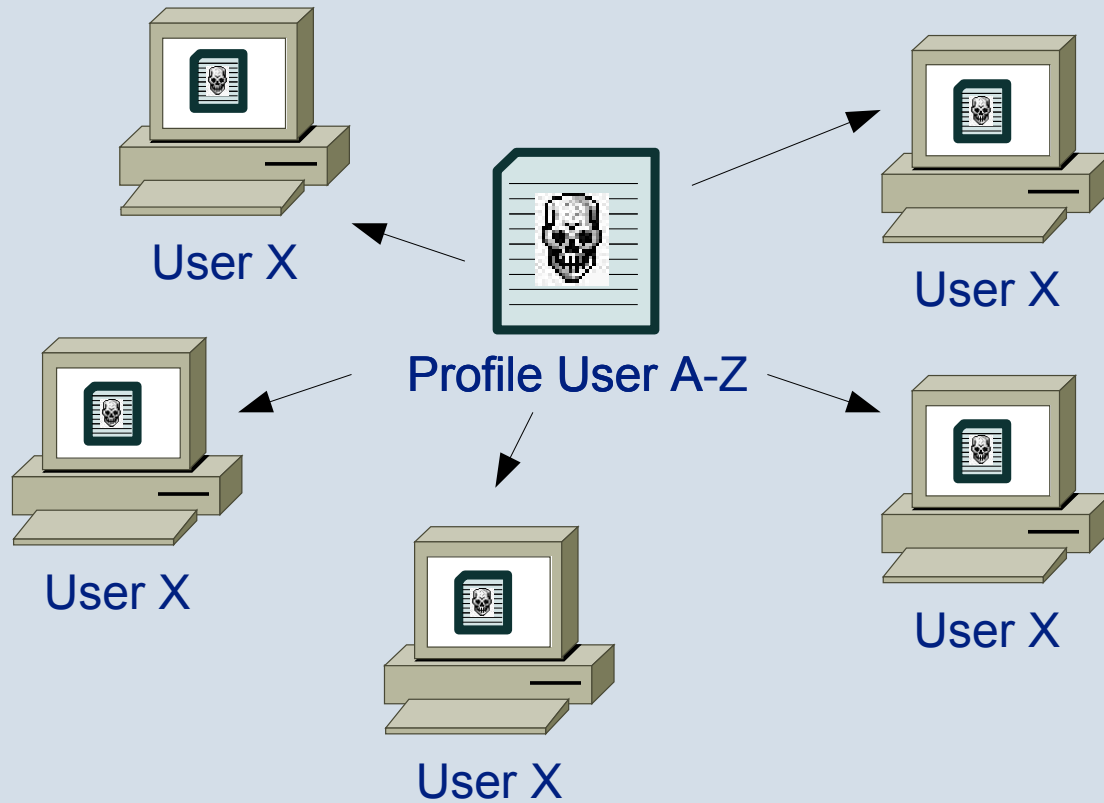
### DDoS

- Based on XSS Worms
- Build "Webbased dynamical Botnets"
  - Javascript means control over the webbrowser
  - Javascript don't have to be static

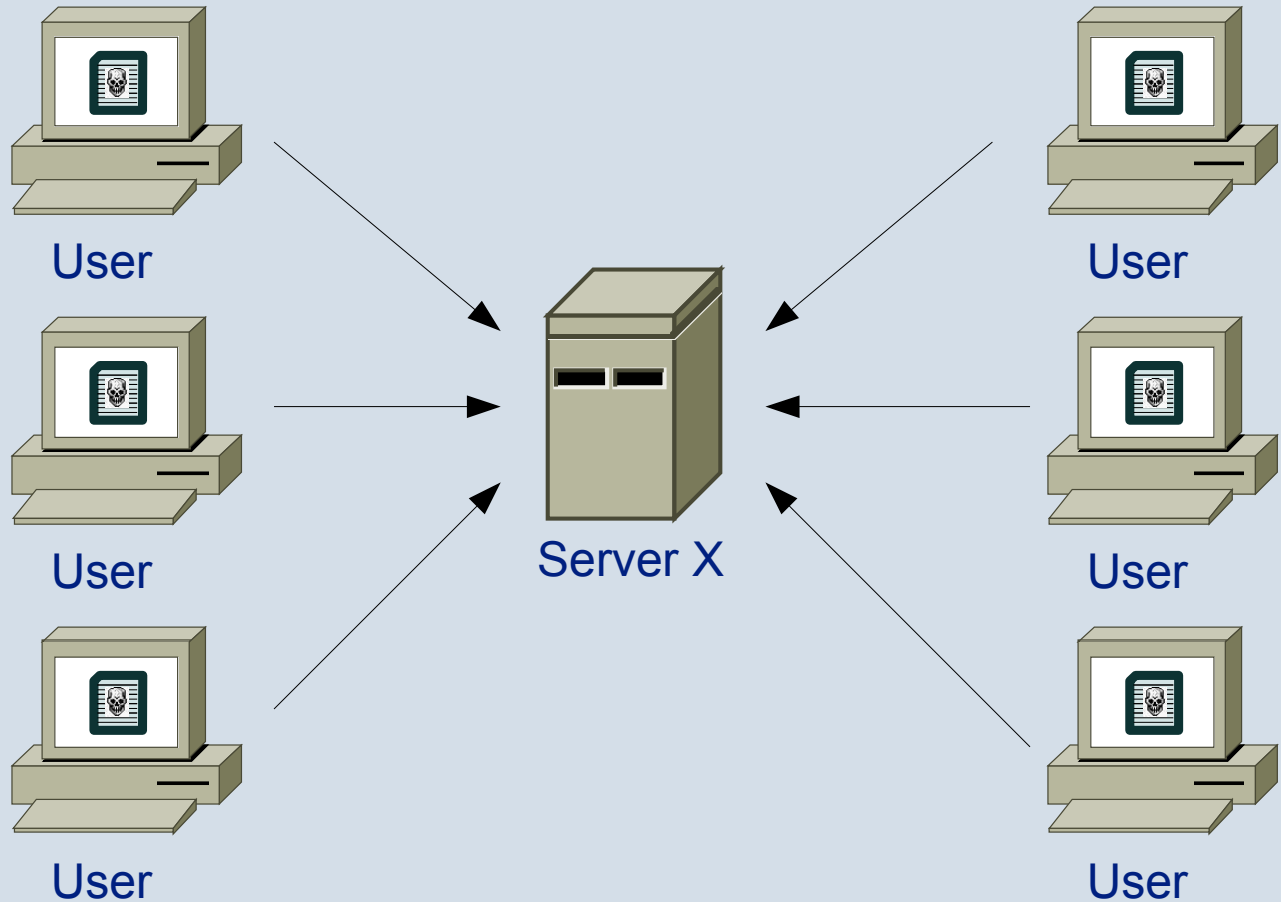
# DDoS



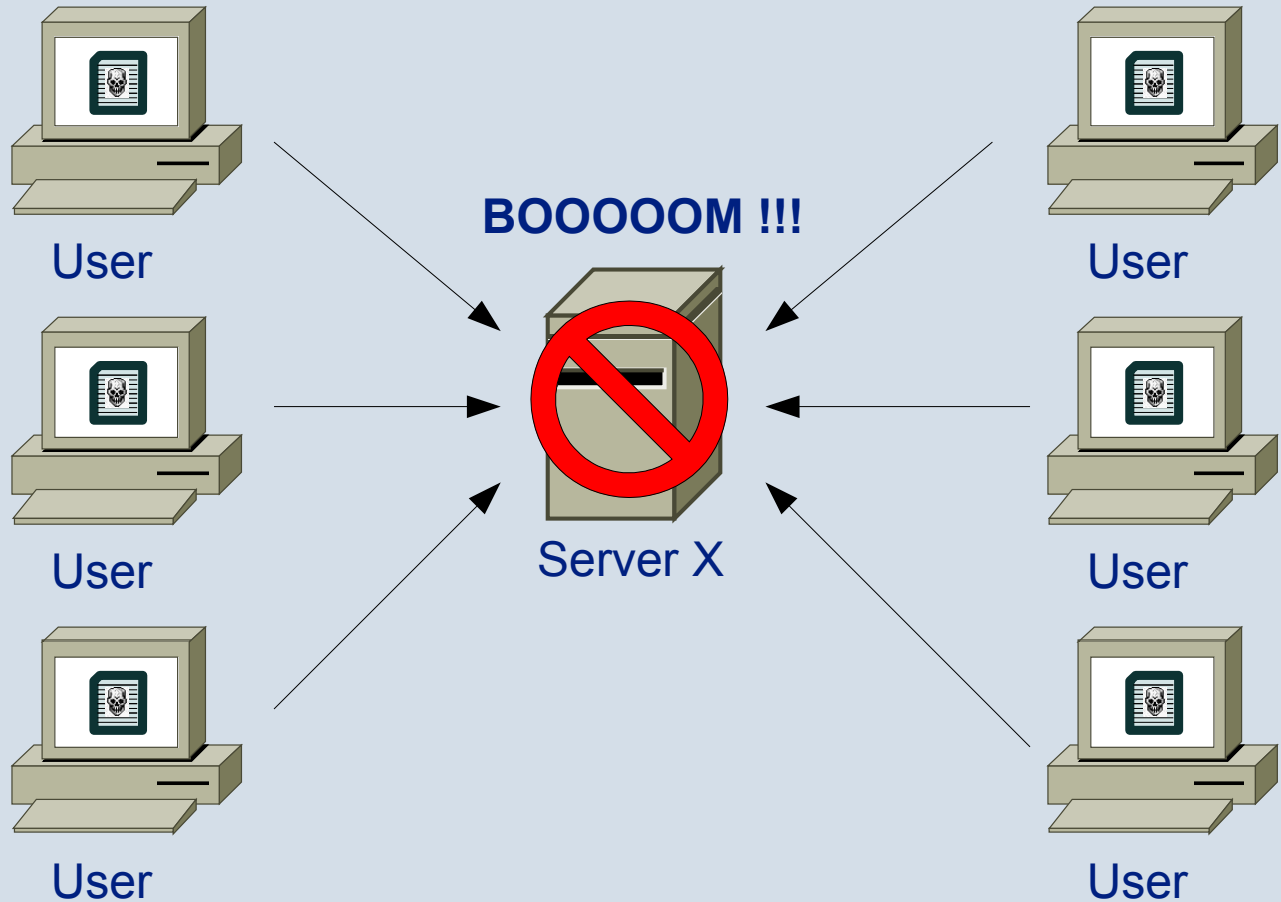
# DDoS



# DDoS

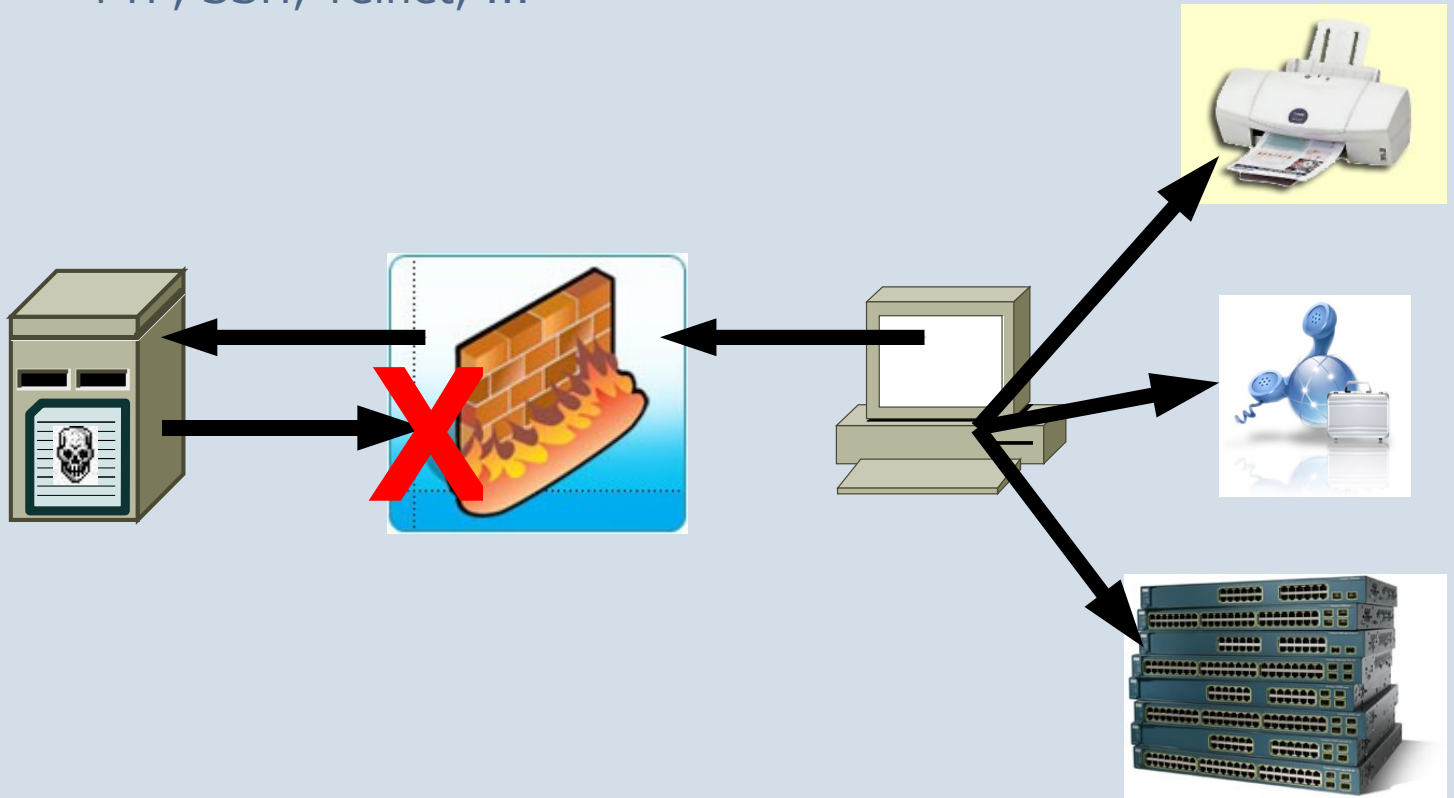


# DDoS



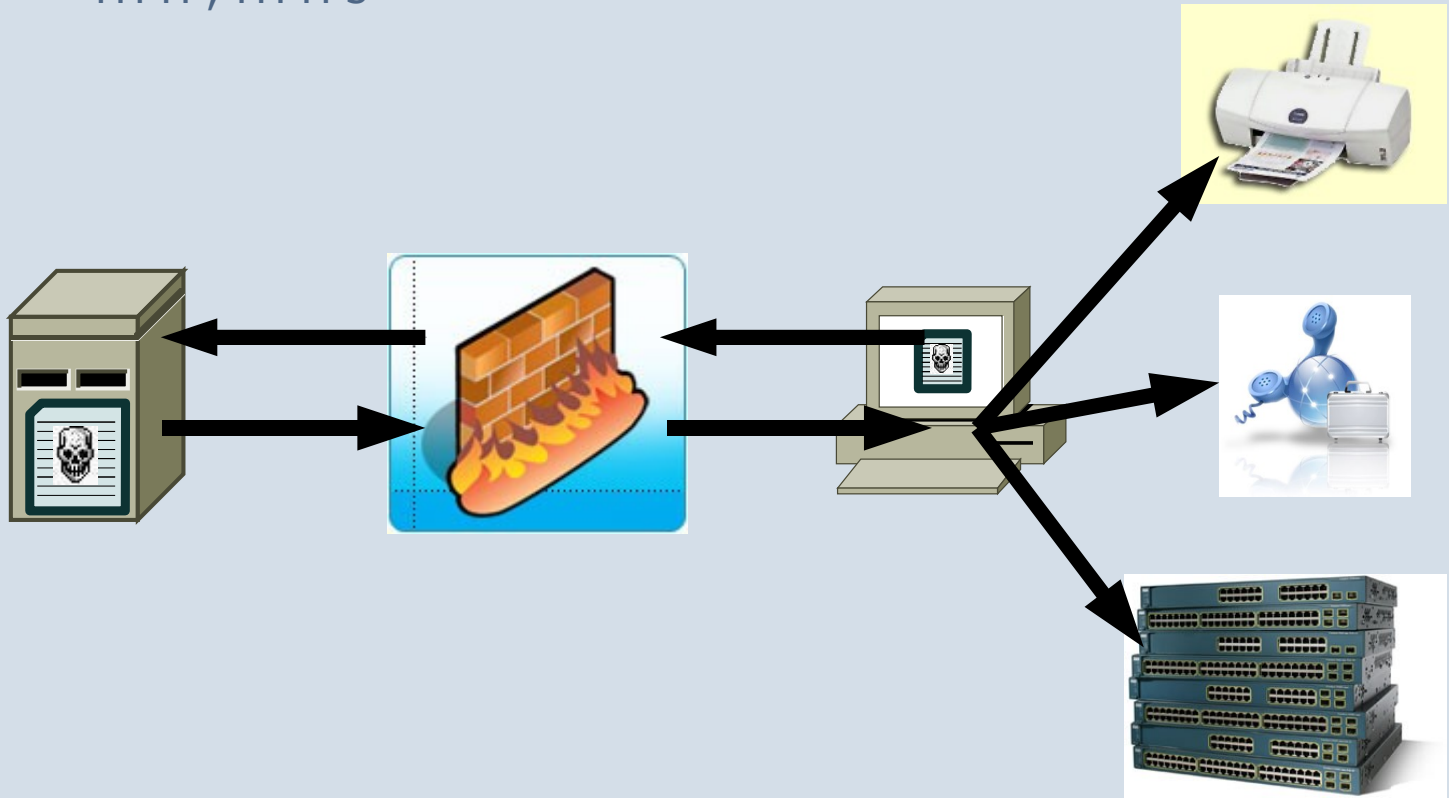
# Webbased Malware

- FTP, SSH, Telnet, ...



# Webbased Malware

- HTTP, HTTPs



## Webbased Malware

- Webbased Keylogger
  - Demo
- Webbased Remote Controll
  - XSS Proxy
- Intranet crawler
  - Intranet Webapplications
- Javascript Portscanner
  - Demo

## **Break into Network Hardware**

- Also nice for Intranet platforms
- Demo

## SEO-Hacking

- Old Method
  - XSS (a Link) in a search variable over GET
  - Linking to the URL with the XSS
  - Get a Backlink
  - (afaik still working except Google)
- New Methode
  - Using the DOM for injecting new Links
  - Permanent Links
  - Using tinyurl.com

## Marketing Research

- Tracking Users
- Get Users History
  - Demo

## XSS Guessing

- Known Variable Attack (KVA)
  - If you have the source
  - If you get some information about the attack vectors
  - Normal Attack-Type on XSS vulnerabilities
- Unknown Variable Attack (UVA)
  - If you don't have the source
  - New Attack-Type on XSS vulnerabilities
  - Find new vulnerabilities
- Demo

## Conclusion

- At the moment Firewalls can't really block XSS
- The possibilities and risks of Javascript are underestimated
- There will be much more attack vectors as we can expect today
- Security will become much more important for webdevelopers
- Security professionals will have much fun ;)



**Thanks for your Attention**

For more information:  
[sven.vetsch\\_at\\_disenchant.ch](mailto:sven.vetsch_at_disenchant.ch)