

# FIT 2093

## Introduction to cyber security

**Space**

Forward

**Right, Down, Page  
Down**

Next slide

**Left, Up, Page Up**

Previous slide



**MONASH**  
University

Open presenter

**H**

console

Toggle this help

# Hacking

**Space**

Forward

**Right, Down, Page  
Down**

Next slide

**Left, Up, Page Up**

Previous slide

**P**

Open presenter  
console

**H**

Toggle this help

# What is hacking?

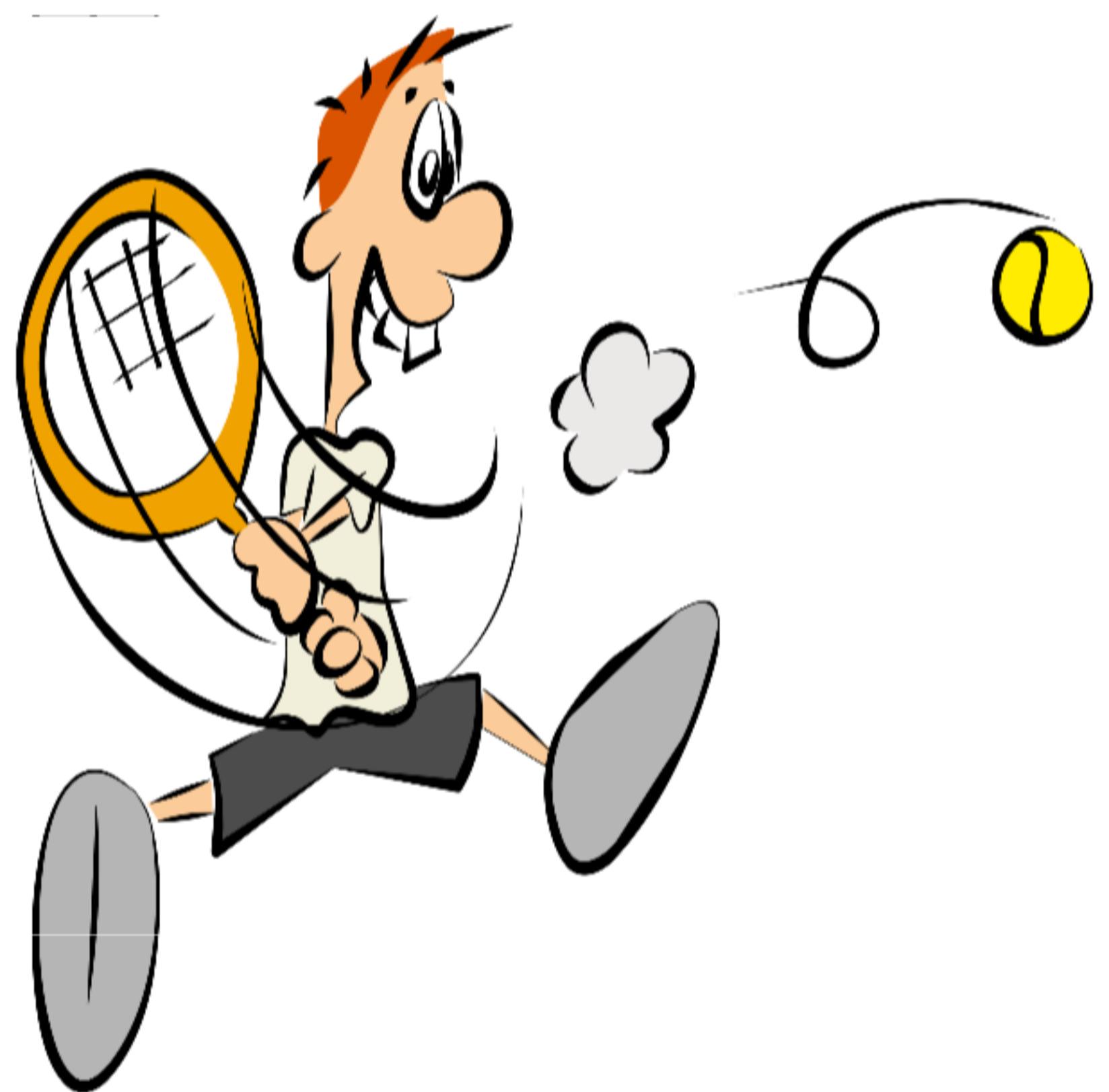
[https://youtu.be/ss47ffaqA\\_o](https://youtu.be/ss47ffaqA_o)

Merriam Webster Dictionary:  
1. One that hacks



Merriam Webster Dictionary:

2. a person who is inexperienced or unskilled at a particular activity: "a tennis hacker"



Merriam Webster Dictionary:  
3. an expert at programming and solving problems  
with a computer

Merriam Webster Dictionary:

4. a person who illegally gains access to and sometimes tampers with information in a computer system



Note that hacking is not the same as cyber-crime

future  tense THE CITIZEN'S GUIDE TO THE FUTURE APRIL 10 2015 7:00 PM

# Eighth-Grader Arrested, Charged With Cybercrimes for Changing Teacher's Desktop Wallpaper

By Lily Hay Newman



Are hacking skills  
essential for attacking IT  
systems?

- YES: Finding weaknesses and developing new attacks can be really difficult.
- NO: Actually exploiting weaknesses can be really easy and just requires general computer skills.

- Penetration testing tools like Metasploit can be used to test computers and networks.
  - Can also be used to learn how to attack.
  - Other frameworks exist for malicious use.
  - Exploits are a business model.

# Look at five areas of hacking

1. Hacking single devices to get access
2. Other attacks on devices and malware
3. Get access to network traffic
4. Gaining access to services
5. Web application hacking

(just a choice...there is more)

# Look at five areas of hacking

1. Hacking single devices to get access
2. Other attacks on devices and malware
3. Get access to network traffic
4. Gaining access to services
5. Web application hacking

# 1. Hacking single devices to get access

- Goal for attacker: Gain full remote access
- Preferably become root (administrator)

# Exploit weaknesses in Software

Examples:

- Buffer Overflow
- Command injection

# Buffer overflow

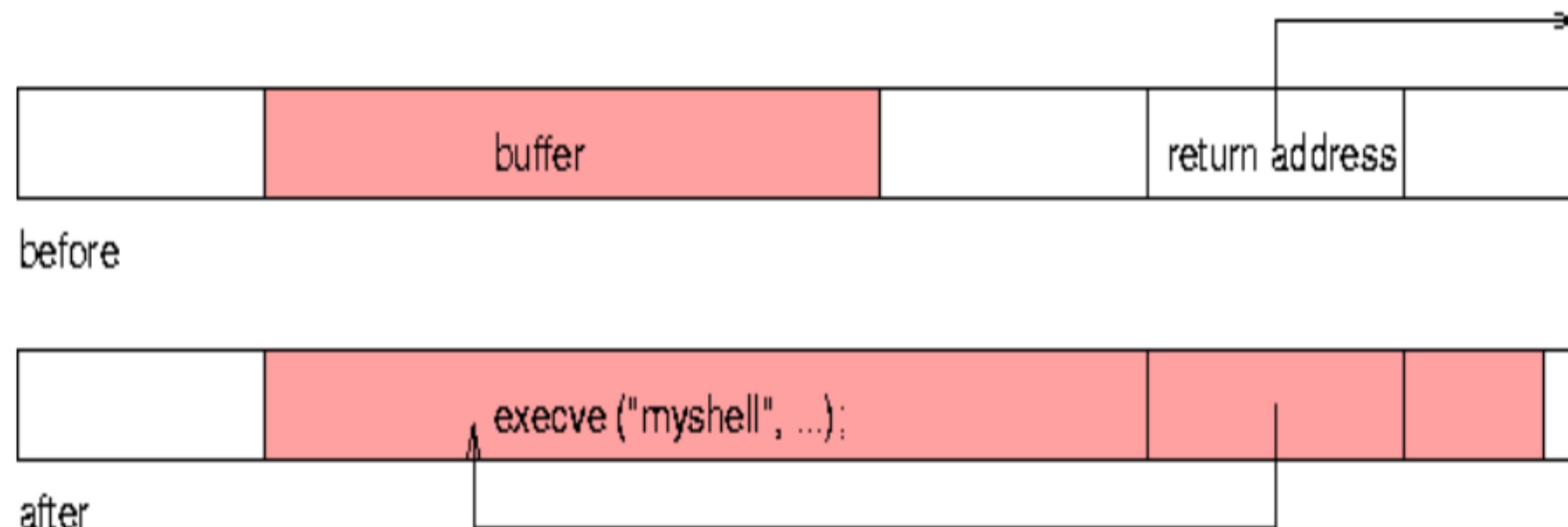


normal subprogram activation record



after buffer overflow

# Example for an exploit



- Buffer overflow can be possible if input is not properly checked.
- Countermeasures do exist (canary, address randomization,...)

# Command Injection

IF an application passes on user input to a shell in a bad way, it can be used to execute arbitrary shell commands with the rights of the application process.

Examples at owasp.org (Open Web Applications Security Project):

[https://www.owasp.org/index.php/Command\\_injection](https://www.owasp.org/index.php/Command_injection)

# Misconfiguration

- Open telnet ports
- Other open ports with vulnerable software
- Wrong access rights
- Missing perimeter protection

# Weak user credentials

- Weak or re-used passwords
- Missing/wrong certificate checks

## Via malicious hardware

- Most prominent example: infected USB devices

 TechRepublic. SEARCH Q

Google CXO Software Cloud Startups More Newsletters Forums Resource Library RSS

SECURITY

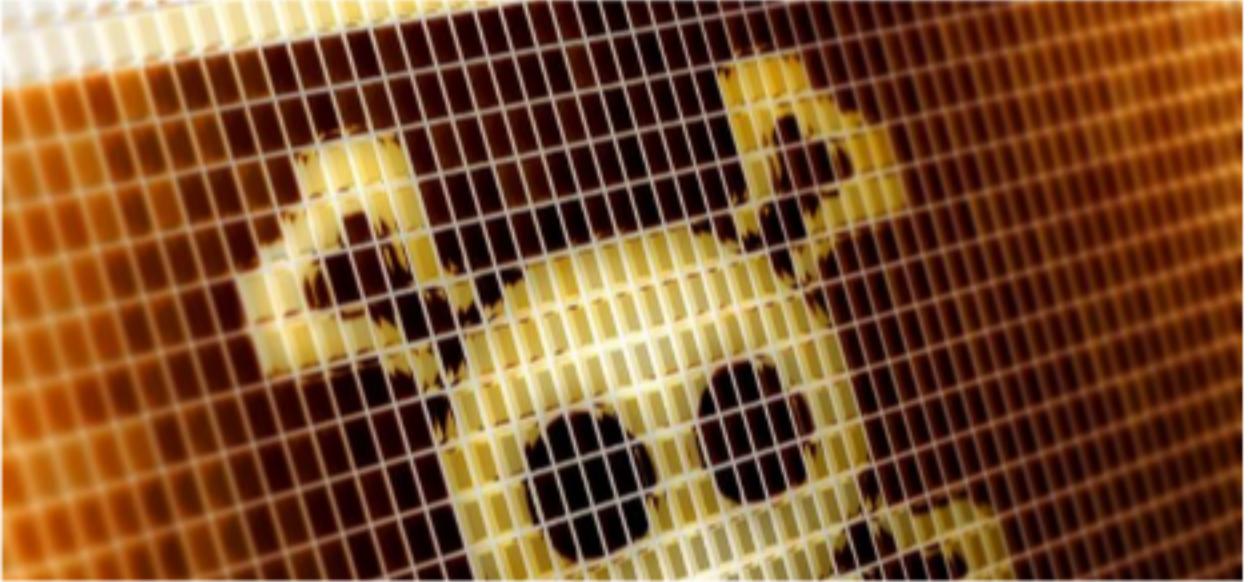
## IBM admits it sent malware-infected USB sticks to customers

In a recent support alert issued by IBM, the company noted that some USB drives that shipped with its Storwize systems contained malicious code.

By Conner Forrest | May 2, 2017, 6:05 AM PST

---

• f in Twitter icon ≡



**FREE NEWSLETTERS, IN YOUR INBOX**

**Tech News You Can Use**  
We deliver the top business tech news stories about the companies, the people, and the products revolutionizing the planet.  
Delivered Daily  
[SUBSCRIBE](#)

**Best of the Week**  
Our editors highlight the TechRepublic articles, galleries, and videos that you absolutely cannot miss to stay current on the latest IT news, innovations, and tips.  
Delivered Fridays

# Social engineering

- Ask for user's credentials
- Convince user to install remote control tool themselves (e.g. "Microsoft" telephone calls)

# What to use access for?

- Establish Botnet (e.g. for large-scale DDoS)
- Illegal activities
- Steal information/data
- Backdoor to a network, used to attack other devices/services in the network
- Install malware
- Sabotage
- Hacktivism

# How to prevent?

- Secure software development
- Install current patches
- Reduce software/services installed/running
- Proper perimeter protection
- Monitor network traffic (also after attack)
- Educate users
- Disable ports (hardware and network)

# Look at five areas of hacking

1. Hacking single devices to get access
2. Other attacks on devices and malware
3. Get access to network traffic
4. Gaining access to services
5. Web application hacking

Often, malware does not need to give someone full control over the computer.

It can spread directly via Network, USB, e-mail, active Web content, etc.

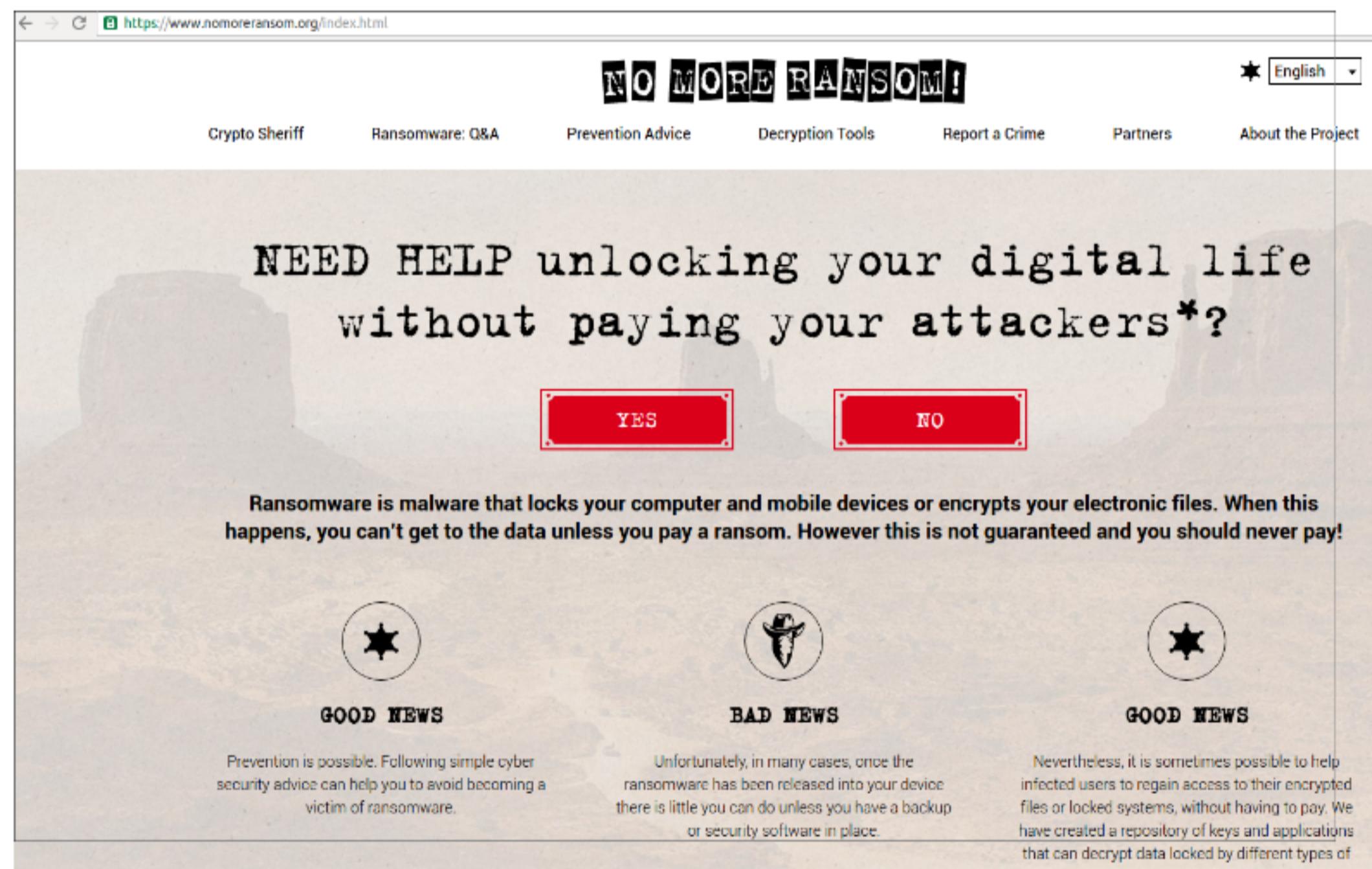
or attacker find alternative ways of distribution, e.g. links in e-mail

# Examples for malware

- Banking malware
- Ransomware
- Spyware
- Botnets
- Adware

Note that these do not need full access to the computer.

# www.nomoreransom.org



The screenshot shows the homepage of the No More Ransom! website. The URL in the address bar is <https://www.nomoreransom.org/index.html>. The page title is "NO MORE RANSOM!". The navigation menu includes links for "Crypto Sheriff", "Ransomware: Q&A", "Prevention Advice", "Decryption Tools", "Report a Crime", "Partners", and "About the Project". A language selector shows "English". The main headline reads: "NEED HELP unlocking your digital life without paying your attackers\*?". Below this are two red buttons labeled "YES" and "NO". A descriptive text block states: "Ransomware is malware that locks your computer and mobile devices or encrypts your electronic files. When this happens, you can't get to the data unless you pay a ransom. However this is not guaranteed and you should never pay!". The page features three circular icons: one with a star labeled "GOOD NEWS", one with a lock labeled "BAD NEWS", and one with a star labeled "GOOD NEWS". Text under the "GOOD NEWS" icons provides information on prevention and decryption tools.

https://www.nomoreransom.org/index.html

NO MORE RANSOM!

Crypto Sheriff Ransomware: Q&A Prevention Advice Decryption Tools Report a Crime Partners About the Project English

NEED HELP unlocking your digital life without paying your attackers\*?

YES NO

Ransomware is malware that locks your computer and mobile devices or encrypts your electronic files. When this happens, you can't get to the data unless you pay a ransom. However this is not guaranteed and you should never pay!

GOOD NEWS

Prevention is possible. Following simple cyber security advice can help you to avoid becoming a victim of ransomware.

BAD NEWS

Unfortunately, in many cases, once the ransomware has been released into your device there is little you can do unless you have a backup or security software in place.

GOOD NEWS

Nevertheless, it is sometimes possible to help infected users to regain access to their encrypted files or locked systems, without having to pay. We have created a repository of keys and applications that can decrypt data locked by different types of

# Other attacks

## Example 1/3

- Pod slurping

Attach a storage device (e.g IPod) to a computer and slurp large amounts of data.

Prevention: Don't let anyone just attache devices.

# Other attacks

## Example 2/3

- Attacks via DMA

Direct memory access DMA can be used to read arbitrary parts of memory

Prevention: Don't let anyone just attach devices.

# Other attacks

## Example 3/3

- Physical (hard-disk access)

With physical (temporary) access, one can directly read from the hard-disk (or write to it) without being logged in.

Prevention: Disk encryption. Self-encrypted disks.

# Look at five areas of hacking

1. Hacking single devices to get access
2. Other attacks on devices and malware
3. Get access to network traffic
4. Gaining access to services
5. Web application hacking

All unencrypted network traffic can be extracted through:

- access to the Ethernet cable
- sniffing wireless traffic
- at each switch/router on the route

Encryption can still leave access to full data on other layers or to meta-data (depending on the layer):

- Wifi encryption is only between client and access point
- TLS only encrypts above transport layer. IP addresses etc. are still visible
- TLS for mail leaves content on mail-server unencrypted
- S/Mime and PGP encrypt mail content, but not headers

Unprotected traffic can also be manipulated/changed!

## IP Address spoofing:

- hide the identity of a sender
- pretend to be another computer (network) for impersonation
- bi-directional traffic difficult, can mainly disrupt

MAC Address spoofing:

- Circumvent MAC-based network access control
- Bring foreign devices in a network and then start attacks, sniff traffic

In some countries, laws require full (sometime real-time) network access for police and secret services!  
What can possibly go wrong?



# Look at five areas of hacking

1. Hacking single devices to get access
2. Other attacks on devices and malware
3. Get access to network traffic
4. Gaining access to services
5. Web application hacking

# Goals for attacking service access

- Criminal activities / Money (banking)
- Identity theft (various services)
- Damage reputation (social networks, personal homepages, e-mail)
- Part of a larger (targeted) attack

# Phishing

- Most successful and frequent.
- Usually via e-mail. Lure users into clicking on a link to a fake website.
- Users provide their credentials on the fake website.
- Attackers can gain full access to service.

Prevention: Two-factor authentication

# Phishing has many variants

- Smishing (via SMS)
- Spear phishing (targeted phishing)
- Combinations with social engineering
- For virtually all types of services

# Look at five areas of hacking

1. Hacking single devices to get access
2. Other attacks on devices and malware
3. Get access to network traffic
4. Gaining access to services
5. Web application hacking

# Web applications

- Interface between users and web-servers
- Dynamic web-pages generated at the server side
- Plus script code executed in the client's Web browser
- Support key business processes
- Often data-base driven

# Vulnerable to various attacks

There are enforcement mechanisms for security policies.

But, there is:

- Cross-site scripting
- SQL injection
- Session hijacking
- Browser/server vulnerabilities
- Third-party content (JavaScript, Flash, ActiveX,...)

# What is HTTP?

- Hypertext Transfer Protocol
- Client requests data from a resource (server)
- Two HTTP request methods: GET (request data) and POST (submit data)
- In principle stateless

# State can be created (for sessions)

- Cookies
- TLS session
- Numbers in URLs

# Server-side technologies

- Scripting languages (PHP, VBScript, Perl)
- Web application platforms ASP.Net, Node.js  
(for server-side Java-script), Ruby on Rails, Java
- Various web servers (Apache, IIS, etc.)
- Databases (Oracle, SAP, MS-SQL, MySQL,  
various NoSQL databases)
- Soap-based web-services / XML messages
- others

# Client-side technologies

- Different browsers
- HTML -> HTML5, CSS, JavaScript
- Extensions (Flash :-/, Java, ActiveX, etc.)
- Others

# How to attack?

1. Analyze the application
2. Change client-side behaviour, bypass client-side controls (scripts, change URLs, cookies, etc.)
3. Attack authentication (if necessary) / Attack session management
4. Attack data stores
5. Analyse and attack back-end components and APIs
6. Attacking users

# 1. Analyze the application

- Spider the application. Can we find the directory structure, files, etc. Create a map
  - Find hidden content
  - Find hidden parameters (e.g. debug=true)
  - Identify entry points for user input (URL, forms, uploads, headers, via cookies, out-of-band)
  - Analyse server-side technologies (versions, file extensions, session tokens, look at requests, etc.)
  - Analyse sessions (cookies, etc.)
  - Map attack surface

## 2. Client side behaviour

- The attacker controls the client side
- Change URLs
- Manipulate Cookies
- Change Scripts
- Circumvent client-side controls

### 3. Attack authentication / session management

- Crack passwords, steal credentials (phishing)
- Create/steal session cookies
- Circumvent authentication
- Steal active sessions

## 4. Attack data-bases

- E.g. by SQL Injection (a bit outdated, but still possible here and there)
- Attacks on MongoDB
- more about this topic next week

# 5. Analyse and attack back-end components and APIs

- Simple things like path traversal vulnerabilities can potentially use with big effect
- NODE.js is a great candidate for building insecure APIs...
- NODE.JS applications can interact freely with the operating system without the benefits of a security sandbox!
- Injection vulnerabilities!

OWASP is a great  
resource

[https://www.owasp.org/index.php/Path\\_Traversal](https://www.owasp.org/index.php/Path_Traversal)

# 6. Attacking users

One example of an attack type:

- Cross-site scripting (XSS) attacks

# What is Cross-site scripting?

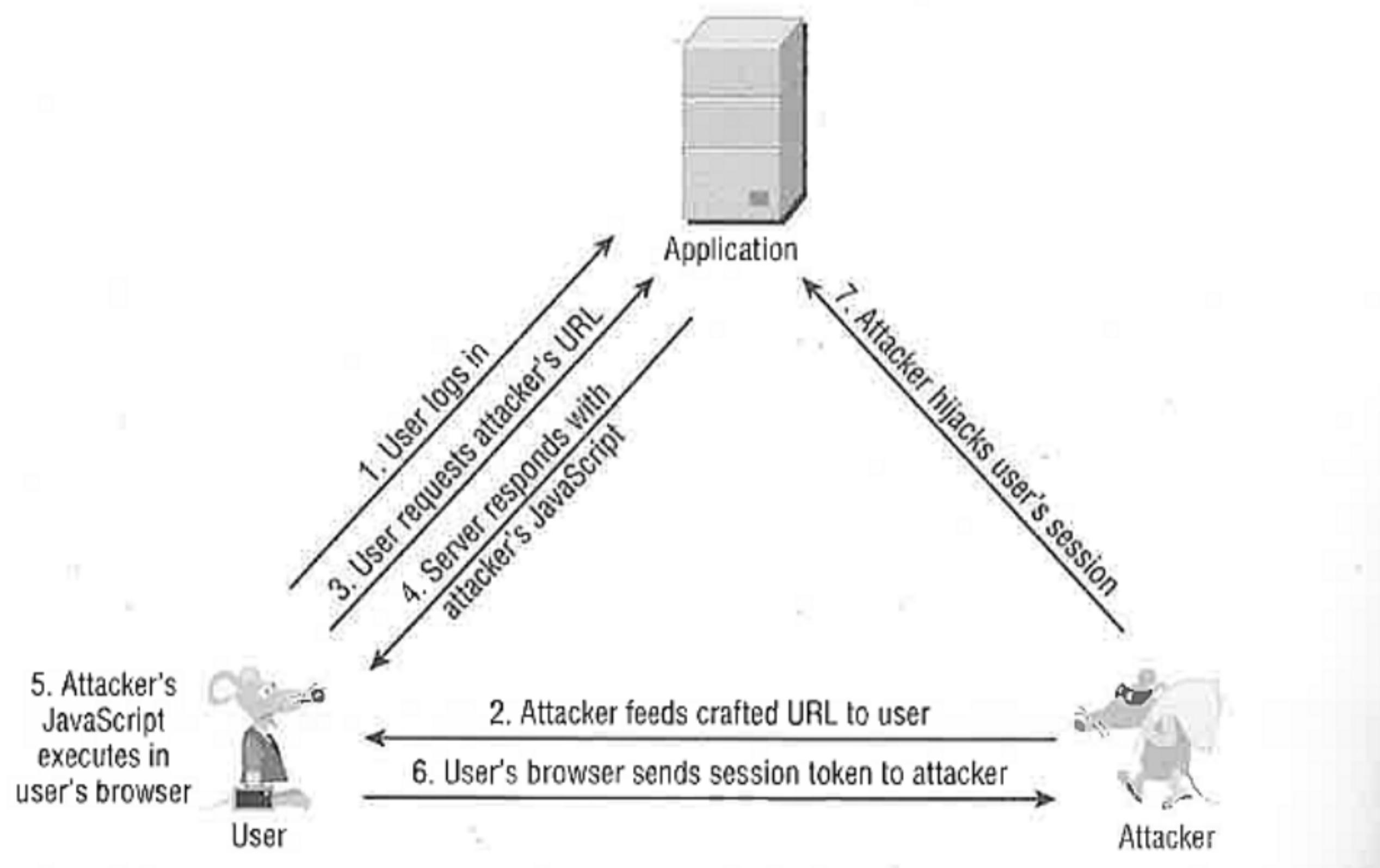
- Usually, browsers don't execute scripts not loaded (directly or indirectly) from the domain of the visited page.
- If attacker can insert own code to be executed is this cross-site.

# How can attacker get script included in the page send from the server?

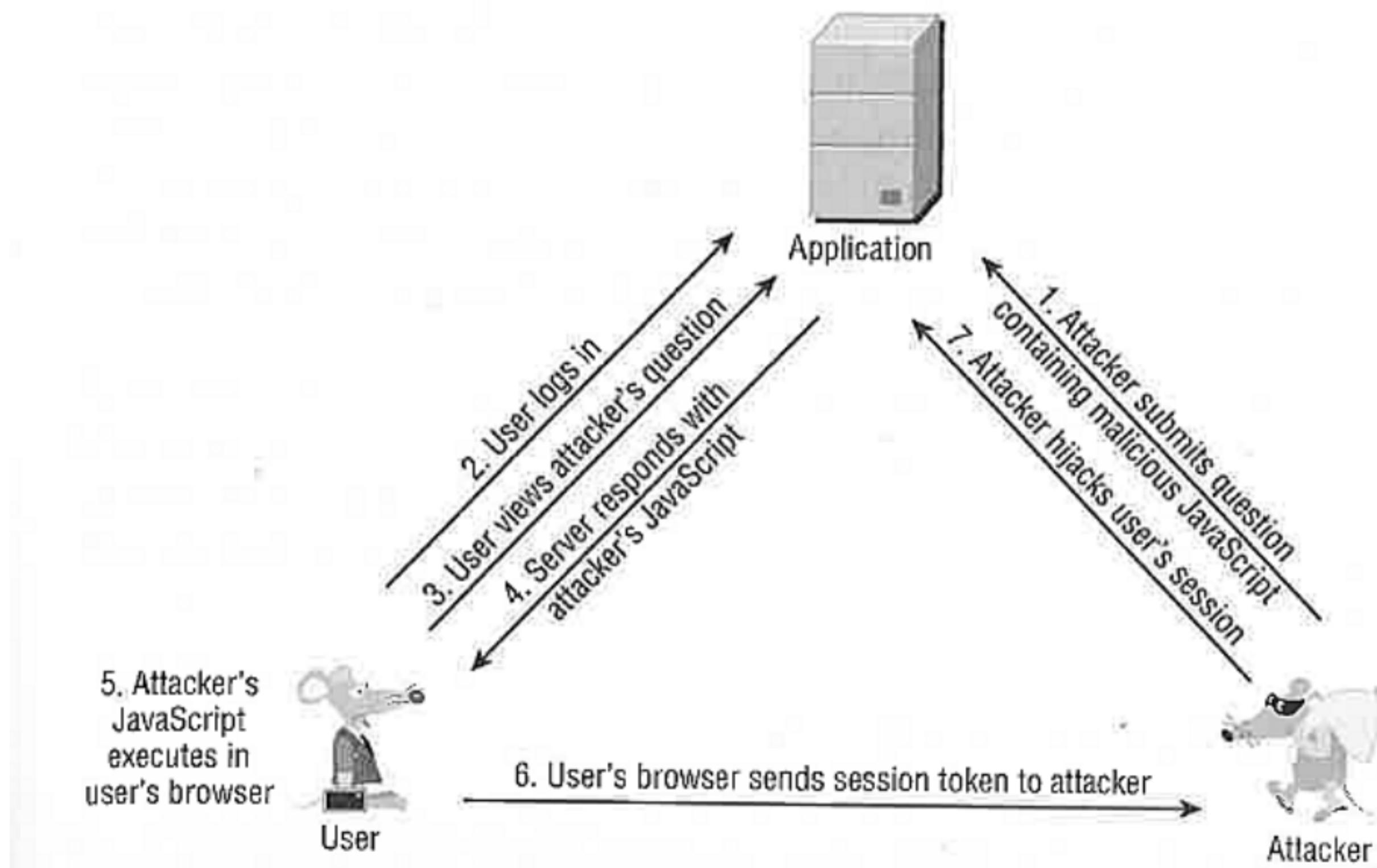
Let's look at two examples:

- Reflected XSS attack
- Stored XSS attack

# Reflected XSS attack



# Stored XSS attack



# How to prevent XSS?

[https://www.owasp.org/index.php/XSS\\_\(Cross\\_Site\\_Scripting\)\\_Prevention\\_Cheat\\_Sheet](https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet)