Security and Privacy

Network and Operational Security Practices

7.05.2019

slide credits: Kirill Nikitn, Ph. Oechslin

EPFL

Outline

- Protecting
  - the network
  - remote access (VPN)
  - the perimeter
  - the workstation
  - the history (logging)
  - data (backups)
- Conclusions and questions

Security breach at Target

source: **Bloomberg**

- Malware installed in Target's security and payment system, steailing details of every credit card used at the company's 1,797 US stores.
- Security alerts on Dec 2, 2014. Targets takes two weeks to react
- Result: 40 million credit card numbers stolen.

Protecting the network

Network Security Practices

- Network segmentation
  - Demilitarized Zone (DMZ) exposes organization's external facing networks to untrusted networks
  - Virtual Local Area Networks (VLANs): network partitioning at layer 2 (Ethernet), for different uses inside a company's network
  - Virtual Routing and Forwarding (VRF): network partitioning at layer 3, running multiple virtual layer 3 (IP) networks
- Secure communication over external network (TLS, IPSec)

Network segmentation

- Break down the network into segments based on system and data classification or into functional zones

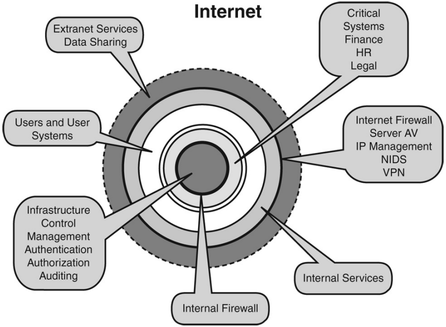
- Access from zone to zone can be managed by access control lists (ACLs) in routers or firewalls

- Mainly addresses two points:
  - Prevents all-at-once compromise of facilities
  - Perimeter defense protects the data center from external threats with little protection against internal threat agents

Network segmentation - Rings

- In classic concentric architecture, access rights to services increase with each level, moving between levels managed by access control

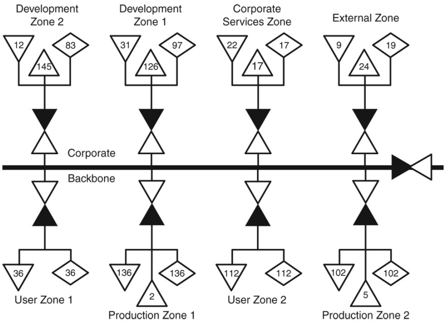
- There my be many machines in one single ring
  - An attack spreading in that ring may create significant damage



source: **NetworkWorld**

Network segmentation - Zones

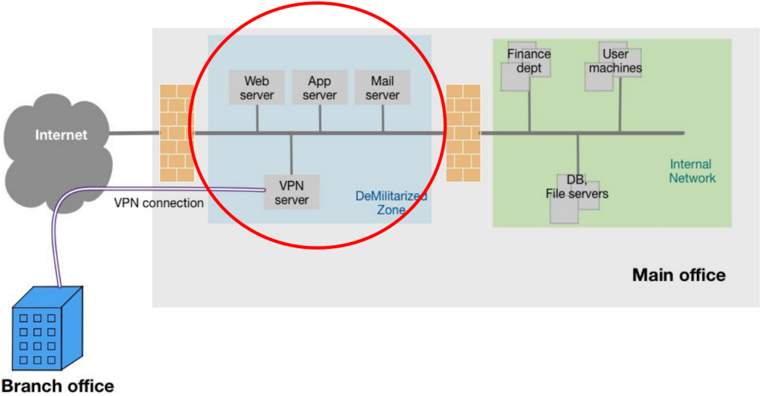
- Creating containment zones aims at stopping attacks from spreading between zones
- Communication between zones goes through firewalls
- The difficulty is creating firewall rules for each case
  ➡ easy to make mistakes



source: **NetworkWorld**

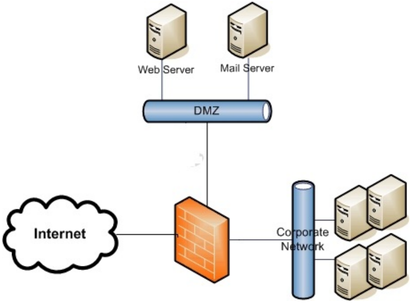
Demilitarized Zone (DMZ)

- A physical or logical subnetwork that contains and exposes an organization's external-facing services to an untrusted network, e.g., Internet.
- An external network node can access only what is exposed in the DMZ
- The most common services in DMZ are web, email, DNS, and FTP servers

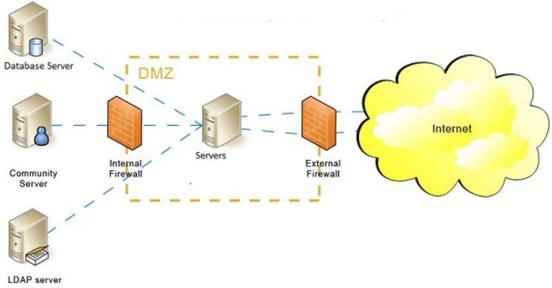


Demilitarized Zone (DMZ)

- two common architectures:



single firewall



dual firewall

source: Mrinal Srivastava **DMZ**

Virtual Local Area Networks (VLANs)

- Virtual LANs allow to run different Local Area Networks over the same layer 2 infrastructure (switches and cables)
- A machine configured in one VLAN will not see traffic of other VLANs even if they share the same switch.
- Typically, switch ports that connect to a wall socket are configured to one VLAN
- Cables that interconnect switches (trunks) carry traffic of all VLANs
- To know to which VLAN a packet belongs, a tag is added to its Ethernet header

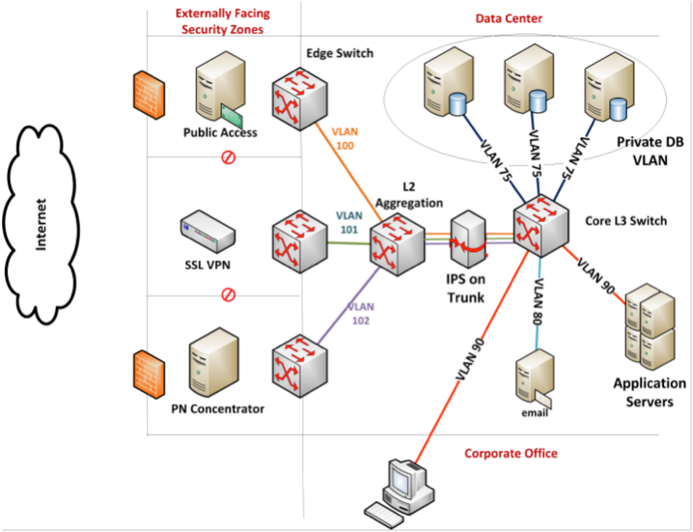
Virtual Local Area Networks (VLANs)

What do “isolated networks” provide?

- Authorized users can “see” only their network segment.
- Flexibility: You can configure to have any VLAN on any wall socket in the building
  - the machines in one network zone don’t have to be located in one physical zone
- Configuration can be dynamic (by machine type, user login, ...)
  - an IP phone is assigned to the telephony VLAN
  - an unknown machine is assigned to guest VLAN
  - a company laptop is assigned to the office VLAN
  - the port of an infected machine can be reassigned to a quarantine network

Virtual Local Area Networks (VLANs)

Possible setup for an organization



source: Infosec Institute

VLAN Attacks

VLAN Hopping enables traffic from one VLAN to be seen by another VLAN:

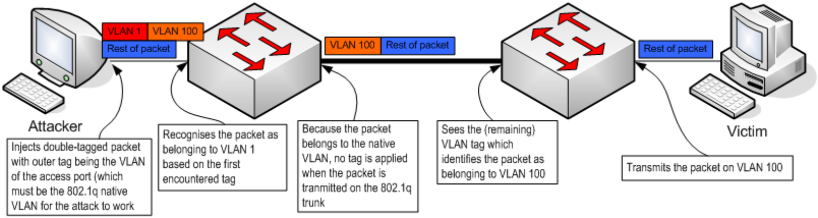
- Switch Spoofing
  - An attacker takes advantage of incorrectly configured switch ports.
    - The attacker pretends to be a switch by emulating config messages and forming a trunk with a legitimate switch.
  - A defense is to disable auto port trunking (switch-to-switch connection) and set it manually.

VLAN Attacks

VLAN Hopping enables traffic from one VLAN to be seen by another VLAN.

- Double tagging
  - Most switches perform only one level of 802.1Q de-encapsulation (tag removing),
  - The attacker adds two tags to his packets
  - The first switch removes the first tag if it corresponds to the default VLAN and forwards the packet
  - The second switch sees the second tag and sends the packet to the corresponding ports
  - A defense is to ensure that the native VLAN of the trunk ports is different from the VLAN of any user ports.

Double Tagging Attack



Injects double-tagged packet with outer tag being the VLAN of the access port (which must be the 802.1q native VLAN for the attack to work

Recognises the packet as belonging to VLAN 1 based on the first encountered tag

Because the packet belongs to the native VLAN, no tag is applied when the packet is tranmitted on the 802.1q trunk

Sees the (remaining) VLAN tag which identifies the packet as belonging to VLAN 100

Transmits the packet on VLAN 100

source: **packetlife**

Protecting remote access

Virtual Private Networks

Virtual private networks

- A VPN extends a private network over a public network

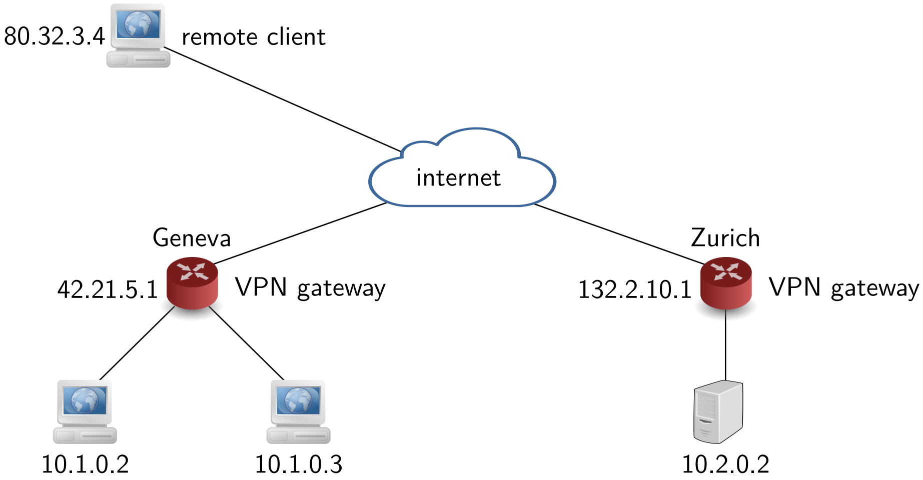
- Encryption and encapsulation keep the network private
  - encryption for confidentiality
  - encapsulation (putting an IP packet inside another IP packet) for transporting packets of the virtual network over the public network

- A network of machines can access a VPN through a VPN gateway

- A single machine can run a VPN software to create a virtual interface
  - this interface has an IP address in the VPN network

- Before a packet is sent over the public network, it is encrypted and encapsulated with an IP header with the public addresses

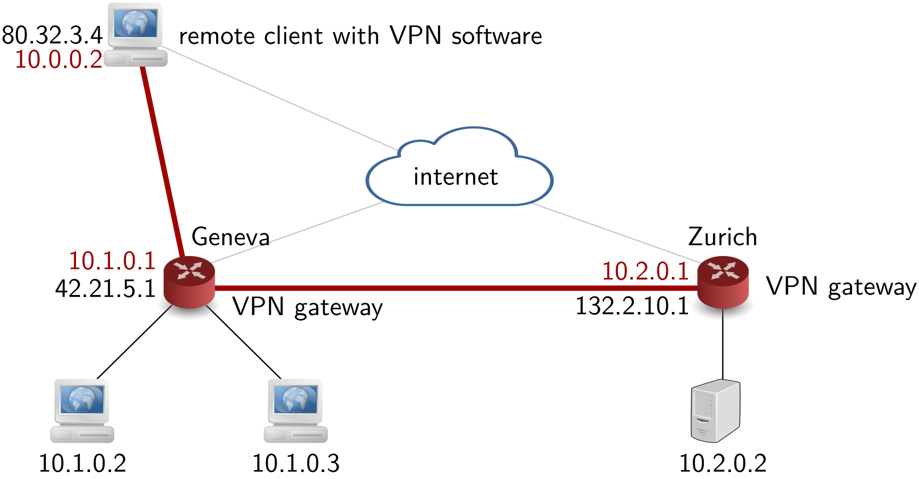
VPN: physical view



- Physical addresses are needed to go from remote to Geneva and Zurich

data	src 80.32.3.4	dst 42.21.5.1

VPN: virtual view



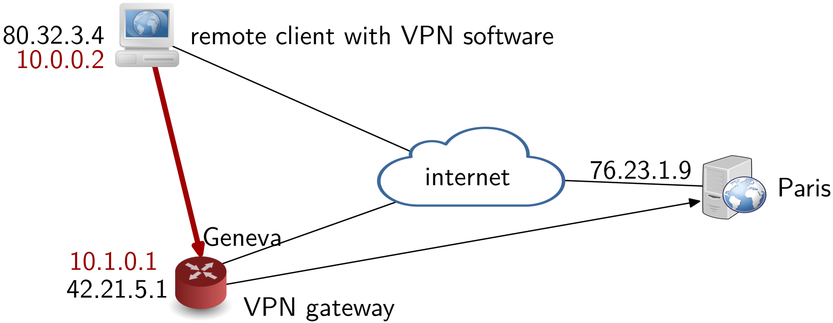
- Packets with virtual addresses are encapsulated in packets with physical addresses:

encrypted data	src10.0.0.2	dst 10.1.0.3	src 80.32.3.4	dst 42.21.5.1

VPN

- Typical protocols
  - IPsec, official IETF standard
  - OpenVPN, open source software, based on TLS
  - Proprietary protocols (e.g. Cisco AnyConnect, Microsoft SSTP)
    - often simpler to configure, potentially less secure
- Typical use
  - Letting remote workers access the internal company network
  - Interconnecting remote sites of a company

VPN for privacy



- User for privacy
  - VPN to another country and access Internet from there
  - Your address is hidden by the one of the VPN gateway
  - ➡ The server in Paris thinks the connections are coming from Geneva
  - ! The VPN provider in Geneva knows who you are and where you are going

Protecting the Perimeter

Firewalls

- Firewalls enforce network level access control
- Typically at the border between Internet and the internal network
  - Larger networks can have multiple zones and firewalls
- Firewalls operate at the network layer
  - They typically look at packet headers (IP, TCP, UDP)
  - They don't know about the application for which the packets are transmitted (mail, web, ...)

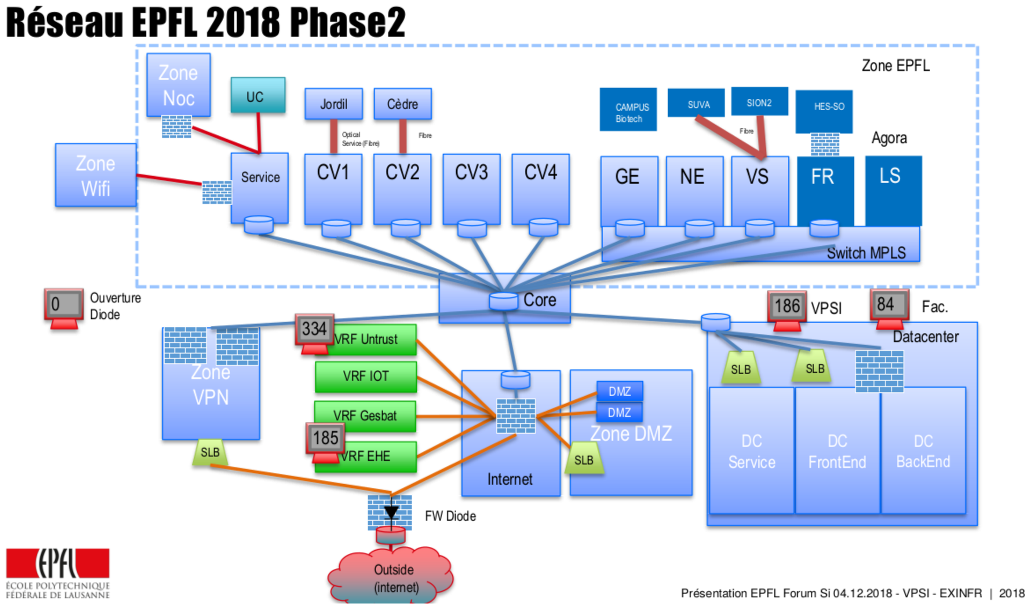
Firewall within the network

- Protecting the network with firewalls at the perimeter is not enough
  - watermelon model (hart outside, squishy inside)

- Propagation of attacks is prevented by segmenting the internal network

- Example: EPFL network segmentation
  - Most machines are in the generic ‘EPFL’ zone
  - All machines accessible from Internet must be located in the ‘untrust’ zone
  - Machines in ‘untrust’ can not connect to machines in ‘EPFL’
    - but the opposite is true
  - The data about your grades and my salary are behind a firewall in the back-end of the ‘Data Center’ zone.
  - In general, each zone can only connect to zones of lesser security
    - remember Mandatory Access Control?

Firewalls & Zones EPFL

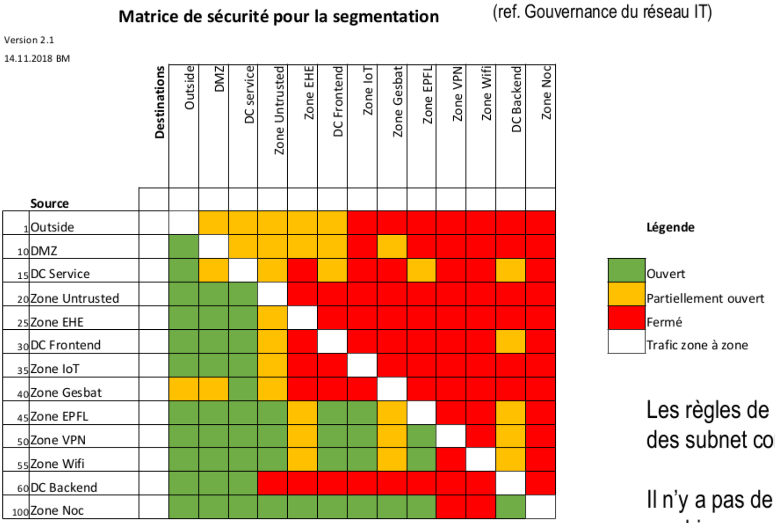


Réseau EPFL 2018 Phase2

Présentation EPFL Forum Si 04.12.2018 - VPSI - EXINFR | 2018

source: **EPFL projet de segmentation (work in progress)**

Firewalls & Zones EPFL



Matrice de sécurité pour la segmentation (ref. Gouvernance du réseau IT)

Version 2.1

14.11.2018 BM

Source		Destinations	Outside	DMZ	DC service	Zone Untrusted	Zone EHE	DC Frontend	Zone IoT	Zone Gesbat	Zone EPFL	Zone VPN	Zone Wifi	DC Backend	Zone Noc
1	Outside		⬜	🟨	🟨	🟨	🟨	🟨	🟥	🟥	🟥	🟥	🟥	🟥	🟥
10	DMZ		🟩	⬜	🟨	🟨	🟨	🟨	🟥	🟨	🟥	🟥	🟥	🟥	🟥
15	DC Service		🟩	🟨	⬜	🟨	🟥	🟨	🟥	🟥	🟨	🟥	🟥	🟨	🟥
20	Zone Untrusted		🟩	🟩	🟩	⬜	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
25	Zone EHE		🟩	🟩	🟩	🟨	⬜	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
30	DC Frontend		🟩	🟩	🟩	🟨	🟥	⬜	🟥	🟥	🟥	🟥	🟥	🟨	🟥
35	Zone IoT		🟩	🟩	🟩	🟨	🟥	🟥	⬜	🟥	🟥	🟥	🟥	🟥	🟥
40	Zone Gesbat		🟨	🟨	🟩	🟨	🟥	🟥	🟥	⬜	🟥	🟥	🟥	🟥	🟥
45	Zone EPFL		🟩	🟩	🟩	🟩	🟨	🟩	🟩	🟨	⬜	🟥	🟥	🟨	🟥
50	Zone VPN		🟩	🟩	🟩	🟩	🟨	🟩	🟩	🟨	🟩	⬜	🟥	🟨	🟥
55	Zone Wifi		🟩	🟩	🟩	🟩	🟨	🟩	🟩	🟨	🟩	🟥	⬜	🟨	🟥
60	DC Backend		🟩	🟩	🟩	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	⬜	🟥
100	Zone Noc		🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟥	🟥	🟩	⬜

Légende

🟩	Ouvert
🟨	Partiellement ouvert
🟥	Fermé
⬜	Trafic zone à zone

Les règles de sécurités s'appliquent à des subnet composant les zones.

Il n'y a pas de règles spécifiques à une machine.

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Présentation EPFL Forum Si 04.12.2018 - VPSI - EXINFR | 2018

source: **EPFL projet de segmentation (work in progress)**

Types of firewalls

- Stateless firewall
  - Looks at packet meta-data, without any context
    - src/dst IP addresses, src/dst TCP/UDP ports, TCP flags, ...
  - Does not look at data
  - Example:
    - Allow only SSH, HTTP and HTTPS traffic
    - Allow only outgoing connections (the TCP ACK flag is absent in the TCP packets that establish a connection):

src	**dst**	**protocol**	**port**	**ACK**	**rule**
internal	external	TCP	22,80,443	any	allow
external	internal	TCP	22,80,443	yes	allow
any	any	any	any	any	deny

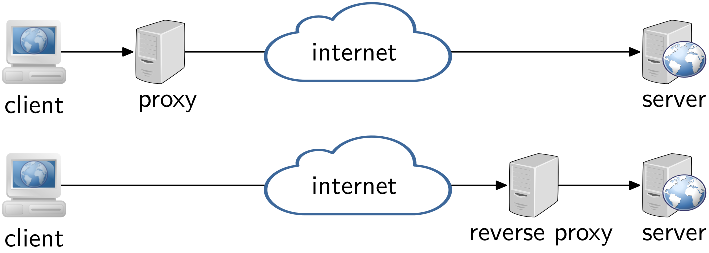
- Principle of default deny
  - The last rules denies all traffic that has not explicitly been allowed before

Types of firewalls

- Stateful firewall
  - Keeps information about traffic it has seen
  - Can make smarter decisions than a stateless firewall
    - e.g. only allow an incoming DNS response (UDP port 53) if we have seen an outgoing DNS request before
    - e.g. only allow incoming TCP traffic if we have seen a connection establishment before

Proxies

- Proxies operate at the application level
  - they act as a server to the client and as a client to the server



- (direct) proxies protect our users when they access servers on the Internet

- reverse proxies protect our servers when accessed by users from the Internet

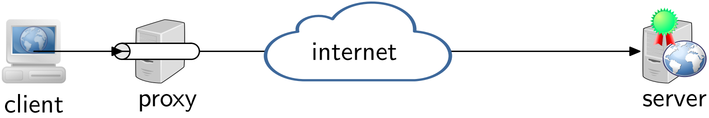
Web proxies

- Web proxies protect the users by
  - analyzing all data downloaded from the web with anti-virus software
    - e.g. webmail attachments
  - blocking access to dangerous sites
- The browsers of the users must be configured to use the proxy
- the firewall can be configured to only allow the proxy to access web sites

  ➡ users can't surf then net if they don't go through the proxy

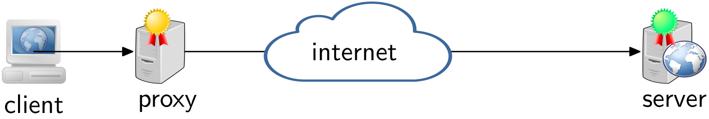
Web proxies for HTTPS

- by default, HTTPS traffic is encrypted from the client to the server.
  - the proxy forward the TLS handshake between the client and the server
  - it just acts like a tube
  - the client receives the original certificate
  - the proxy only sees encrypted traffic
    - it can not filter dangerous data



Web proxies for HTTPS

- The proxy can be configured to intercept HTTPS traffic
  - rather than forwarding the handshake, it pretends to be the server
    - to do this it generates fake certificates
    - the browser sees the fake certificate 🏅 and wants to generate an error
    - all browsers of the company must be configured to accept fake certificates from the proxy



Mail gateways (proxies)

- Mail gateways act as proxy and as reverse proxy
  - all outgoing mail is deposited in the mail gateway before being forwarded to the internet
  - all incoming mail is received by the gateway before being stored in user's mailboxes
- Mail gateways typically offer
  - antivirus protection
  - incoming and outgoing spam protection

Web Application Proxy (WAF)

- A WAF is the typical example of a reverse proxy
- It stands in front of your web server and receives the requests from the Internet
  - It analyses the requests, and if it deems them safe, it forwards them to the real server
- WAFs typically offer the following protection
  - blocks any request that seems to contain an attack
    - cross site scripting, SQL injection
  - limit the number of requests to protect against DoS attacks
    - your site will still be unreacheable, but at least you internal server still works.
  - block an IP address for a certain time after detection of an attack
  - carry out the authentication of users
    - so that unauthenticated users can not even interact with your server

WAF example modSecurity

- modSecurity was originally a module for apache web servers
  - it now supports many web servers and reverse proxies
- There is a free open source set of rules (core rule set)
  - it has thousands of rules
    - SQL injections examples:

```
* "@rx (?i)union.*?select.*?from"
* "@rx (?i:sleep\(\s*?\d*?\s*?\)|benchmark\(.*?\,.*?\))" \
```

    - Cross site scripting examples

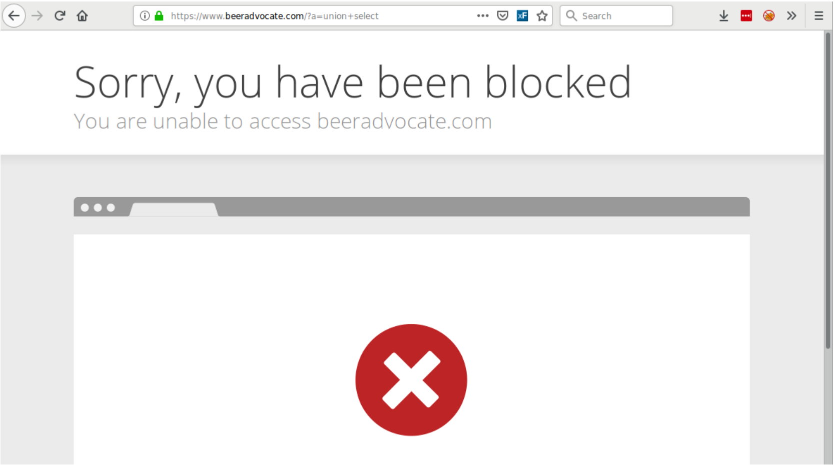
```
* "@rx (?i)[<<]script[^>>]*[>>][\s\S]*?" \
* "@rx (?i)<EMBED[\s/+].*?(?:src|type).*?=" \
```

    - Some rules can be quite complex:

```
/* "@rx (?i:(?:v|&#x?0*(?:86|56|118|76);?)(?:\t|&(?:#x?0*(?:9|13|1
0|A|D);?|tab;|newline;))*(?:b|&#x?0*(?:66|42|98|62);?)(?:\t|&(?:
#x?0*(?:9|13|10|A|D);?|tab;|newline;))*(?:s|&#x?0*(?:83|5
3|115|73);?)(?:\t|&(?:#x?0*(?:9|13|10|A|D);?|tab;|newline;))*(?:
c|&#x?0*(?:67|43|99|63);?)(?:\t|&(?:#x?0*(?:9|13|10|A|D);?|...
```

WAF demo

- Some WAFs can be quite paranoia
  - just add `union+select` anywhere in the URL to have the page blocked



Sorry, you have been blocked

You are unable to access beeradvocate.com

source: **beer advocate**

Intrusion detection systems (IDS)

- An IDS inspects traffic for all applications to detect potential intrusions
  - Generates alerts if it thinks it saw an attack
  - Called an Intrusion Prevention System (IPS) when it also blocks such traffic

- Two technologies
  - Signature-based systems
  - Anomaly-based systems

- Possible issues
  - False positives (too many alarms)
  - False negatives (too many successful attacks)

QUIZ!

- Imagine a situation where
  - the probability that a packet is part of an attack is 1 in a million
  - the false positive rate of your IDS is 0.1% for a single packet
  - the false negative rate is 0.1% for a single packet
  - It is 2am, you receive an alert from your IDS
- What is the chance that you really are under attack:
  - 10%
  - 99.9%
  - 0.1%
  - 50%
- Do you get up or stay in bed ?

Signature based IDS

- Network traffic is compared to signatures from a pattern database
  - like a WAF does for web traffic

- Possible issues
  - requires previous knowledge of an attack to create a signature
    - ➡ false negatives on new attacks
  - matched signature does not always mean attack
    - e.g signature for a Linux vulnerability in traffic for a Windows server
    - ➡ false positives

- Signature examples
  - high number of failed login attempts
  - URLs with extra long parameters (buffer overflow?)
  - Exploiting a specific vulnerability:
    Cisco IOS invalid IKE fragment length memory corruption or exhaustion attempt

Signature based IDS Snort

- Snort is an example of a signature based IDS
- Sniffs traffic in front of your firewall to detect potential attacks
- Sends alarms and/or updates firewall, blocking the attacker
- Large database of free rules maintained by community
- Commercial database of additional rules
- Originally developed by Sourcefire
  - acquired for $2.7Bn to Cisco in 2013
    - Martin Roesch invented Snort in 1998
    - Created his company SourcFire in 2001
    - Went public in 2007
    - Had about 650 employees when acquired by Cisco

Snort Example

- This rule detects an infected Android phone that connects to the attackers server

```
alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS
(
 msg:"MALWARE-CNC Andr.Trojan.Femas variant outbound connection";
 flow:to_server,established; content:"did=";
 http_client_body; content:"/update/upfolder/updatefun.php";
 fast_pattern:only;
 http_uri; content:"Dalvik/"; http_header; content:"Android"; within:25;
 http_header; metadata:impact_flag red, policy balanced-ips drop,
 policy security-ips drop, ruleset community, service http;
 reference:url,blog.lookout.com/blog/2017/02/16/viperrat-mobile-apt/;
 reference:url,securelist.com/blog/incidents/77562/breaking-the-weakest-
 link-of-the-strongest-chain/;
 classtype:trojan-activity; sid:43981; rev:1;
)
```

Anomaly-based Systems

- IDS creates traffic profile during normal operation to calibrate
- During monitoring, it looks for unusual packet
  - e.g. growth in port scans
- Can potentially detect new undocumented attacks
  - might not detect an SQL injection (extra characters in username)
  - might notice a 200MB transfer of password hashes from the database
- Generates a lot of false positives and negatives (anomaly $\neq$ attack)
- Very hot application for machine learning techniques

Protecting the Workstation

Protecting the Workstation

- Prevent exploitable bugs
  - ensure automatic update of the OS
  - ensure automatic update of the applications
- Prevent known malware (on Windows)
  - install antivirus or use built-in Windows Defender
- Prevent unknown malware (on Windows)
  - remember w ^ x from buffer overflows ?
    - you can configure directories such that user can either write or execute files, but not both
      - e.g they write into "my documents"
      - and they execute from "c:\Applications"
      - they will not be able to download and execute malware files
    - stricter: you can make a whitelist of applications that can be run by normal users

Preventing privilege escalation

- Verify that all access rights are correctly set on programs and libraries
- Typical issue: a program that is run by System is writable by any user
  - the attacker replaces the program by a malicious program and reboots the machine
- Typical issue: the hard disk is not encrypted and booting from USB stick is not forbidden:
  - the attacker boots Linux from an USB stick.
  - she can read the hash of the local administrator of the machine
    - if she can crack the hash, she can then log in as admin
    - if not she can use the hash to access the machine remotely

Disabling dangerous features (LLMNR)

- By default Windows uses local name resolution mechanism when DNS resolution fails
  - Link-local Multicast Name Resolution (LLMNR)
  - The machines ask if any machine in the same LAN happens to know the IP address of a host name.

- When you type ‘sushi’ in the address bar of your browser, it doesn’t know if you are searching for pages about raw fish, or if you want to connect to a machine called sushi
  - in order to know, the browser tries to resolve ‘sushi’ with DNS
  - if it fails, it will use LLMNR to ask it neighbors
    - “Does anyobdy know a machine called sushi?”
  - the attacker says “Sushi? That’s me!”

Disabling dangerous features (LLMNR)

- The victim proceed to connect to the attacker's machine
- That machine says that connections are only possible with challenge-response authentication and sends a challenge
- The victim send a correct response to the challenge
- The attacker can now try to bruteforce the password using the challenge and the response

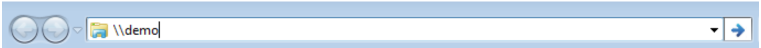
LLMNR demo

- Run responder.py on a Linux machine that can see the traffic of a (virtual) Windows machine:

```
  .----.-----.-----.-----.-----.-----.--|  |.-----.----.
  |   _|  -__|__ --|  _  |  _  |     |  _  ||  -__|   _|
  |__| |_____|_____|   __|_____|__|__|_____||_____|__|
                 |__|

          NBT-NS, LLMNR & MDNS Responder 2.2

  Original work by Laurent Gaffie (lgaffie@trustwave.com)
  To kill this script hit CRTL-C

[+] Poisoners:
   LLMNR                      [ON]
   NBT-NS                     [ON]
   DNS/MDNS                   [ON]

[+] Servers:
   HTTP server                [ON]
   HTTPS server               [ON]
   WPAD proxy                 [OFF]

[+] Generic Options:
   Responder NIC              [usb0]
   Responder IP               [192.168.42.127]
   Challenge set              [1122334455667788]

[+] Listening for events...
```

LLMNR demo

- On the Windows machine, try to open a network location that does not exist



\\demo

```
.   [*] [LLMNR]  Poisoned answer sent to 192.168.42.57 for name demo
    [SMB] NTLMv2-SSP Client   : 192.168.42.57
    [SMB] NTLMv2-SSP Username : os-PC\user
    [SMB] NTLMv2-SSP Hash     : user::os-PC:1122334455667788:491F6D9FAC912904164267F8508C21E1:
    01010000000000008BEBA3ED1A04D501409BC2A2B6D9FCB40000000002000A0073006D00620031003200001001
    40053004500520056004500520032003000300038000400160073006D006200310032002E006C006F00630061
    006C0003002C0053004500520056004500520032003000300038002E0073006D006200310032002E006C006F0
    0630061006C000500160073006D006200310032002E006C006F00630061006C00080030003000000000000000
    0100000000200000979B93F721F1994999DFAB704B63104C162020DBD84D7A63251C5A25E1F7938F0A0010000
    000000000000000000000000000000009001A0063006900660073002F007600610063006100740069006F006E
    000000000000000000
```

- then crack the challenge-response with hashcat

```
$ hashcat -m 5600 logs/SMB-NTLMv2-SSP-192.168.1.247.txt  dict/french -r rules/best64.rule
hashcat (v4.0.1) starting...
[..]
USER::os-PC:1122334455667788:[..]:Maison2
```

Disabling dangerous features (WPAD)

- WPAD (Web Proxy Auto-Discovery) lets a Windows machine find out if it should connect to a proxy to browse the Internet
  - The browser asks “Is there any proxy I should connect to?”
  - The attacker answers “Proxy? Yes, that’s me!”
  - The browser thus connects to the attacker
  - The attacker asks for challenge-response authentication and can brute-force the victim’s password
- Responder.py can also fake WPAD replies.

Protecting History (logging)

Logging

- Keeping audit trails, aka logs is an important part of network security
  - identify security incidents
  - monitor policy violations
  - non repudiation control

- Typical sources of logs
  - Firewalls, proxies and IDS
  - Client & Server machines
  - Mail servers
  - Database applications

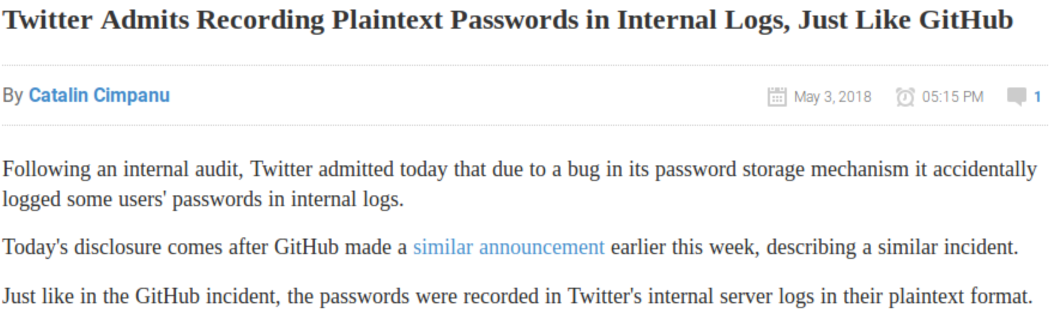
Logs example

```
11.8.12.38 - [16/Jan/2018:01:26:11 +0200]
  "GET /lt/contact.htm HTTP/1.1" 200 2513 "-"
  "Mozilla/5.0 (compatible; YandexBot/3.0; +http://yandex.com/bots)"
14.76.5.24 - [16/Jan/2018:02:19:06 +0200]
  "GET / HTTP/1.1" 200 5073 "http://vilniusquartet.com/"
  "Mozilla/55 (Windows NT 10.0; WOW64; rv:55.0) Gecko/20100101 Firefox/55"
14.76.5.24 - [16/Jan/2018:02:19:06 +0200]
  "GET /css/style.css HTTP/1.1" 200 2480 "http://www.krom.org/"
  "Mozilla/55 (Windows NT 10.0; WOW64; rv:55.0) Gecko/20100101 Firefox/55"
15.220.101.40 - [16/Jan/2018:02:22:13 +0200]
  "GET /css/functions.js HTTP/1.1" 200 1802 "http://www.krom.org/"
  "Mozilla/5.0 (Windows NT 6.1; Trident/7.0; rv:11.0) like Gecko"
132.29.235.184 - [16/Jan/2018:03:56:13 +0200]
  "GET /vvk/ HTTP/1.1" 200 5073 "-"
  "Mozilla/5.0 (iPhone; CPU iPhone OS 9_1 like Mac OS X) AppleWebKit/601.1
   (KHTML, like Gecko) Version/9.0 Mobile/13B143 Safari/601.1"
```

Logs of an Apache web server

Things you should not log

- Passwords!



Twitter Admits Recording Plaintext Passwords in Internal Logs, Just Like GitHub

By Catalin Cimpanu May 3, 2018 05:15 PM 1

Following an internal audit, Twitter admitted today that due to a bug in its password storage mechanism it accidentally logged some users' passwords in internal logs.

Today's disclosure comes after GitHub made a similar announcement earlier this week, describing a similar incident.

Just like in the GitHub incident, the passwords were recorded in Twitter's internal server logs in their plaintext format.

source: **Bleeping computer**

Things you should not log

- Swiss federal act on data protection requires strict security mechanisms for log containing sensitive personal information
  - religious, ideological, political or trade union-related views or activities,
  - health, the intimate sphere or the racial origin,
  - social security measures,
  - administrative or criminal proceedings and sanctions;

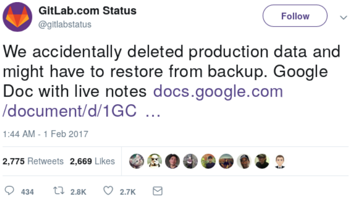
- Basically, the content of potentially private e-mail and Internet access logs can contain sensitive information

- Internet access logs should only be generated in an anonymous way.
  - nominal analysis of Internet access is only allowed if there are tangible signs of abuse

- Mailboxes and logs should be protected against unauthorized access

Protecting Data (backups)

Backups



GitLab.com Status
@gitlabstatus

We accidentally deleted production data and might have to restore from backup. Google Doc with live notes docs.google.com/document/d/1GC …

1:44 AM - 1 Feb 2017

2,775 Retweets 2,669 Likes

434 2.8K 2.7K

source: **Gitlab**

- Timeline
  - 2017/01/31 6pm UTC: Spammers are hammering GitLab's database, causing a lockup.
  - 2017/01/31 10pm UTC: DB replication effectively stops.
  - 2017/01/31 11pm-ish UTC: team-member-1 starts removing db1.cluster.gitlab.com by accident.
  - 2017/01/31 11:27pm UTC: team-member-1 terminates the removal but 300 GB of data is lost.
- They figure out that regular backups are only done once every 24 hours, and some system parts are not backed up at all.
- GitLab manages to restore from a six-hour-old backup but loses all the data submitted after.

Backup types

- Backup types
  - Full: all the data
  - Incremental: only data that has changed since last backup
  - Differential: only data that has changed since last full backup
- 3-2-1 rule:
  - 3 copies of the data
  - on 2 different types of media (e.g disk and tape)
  - 1 stored off-site

Backup content

Backups are made differently depending on the type of data:

- Backup of (virtual) machines: allow to restore a machine without having to re-install and configure all the software
- Backup of databases: databases have their own tools to backup and restore the content of their tables
- File storage: data storage servers can archive every version of every file up to the last backup (e.g. one day).
  - no data is lost if files are accidentally deleted during the day.

Disaster recovery plan

Making backups is not sufficient. We also need

- restoration tests, to check if we are actually able to restore data from backups
- a Disaster Recovery Plan (DRP) that explains in details how to rebuild each system in case of a major failure
  - It is a good practice to keep a copy of the DRP off-site.

Conclusions & Questions

Conclusions

- Security has to be implemented at many different levels
  - important to verify that we didn't forget anything

- There are so many things that can go wrong if there are people with malicious intentions
  - We want to prevent bad things from happening, while letting people do their work
    - principle of least privilege

- Most measures we have seen (Vlans, zones, firewalls, proxies, VPNs) are preventive measures

- Antivirus, WAFs, IDS are detective measures
  - detective measures have a higher operational cost because they generate false positives and possibly false negatives

Conclusions

- Logs and backups do not prevent or detect anything
  - they limit the impact if something bad happens

Questions

- For a given vulnerability (say a buffer overflow in a certain type of web servers)
  - is it better to patch your servers or to add a rule in your IDS to detect and block attempts to exploit this vulnerability?
  - When would it make sense to do both?
- Why do VPNs use encryption ? Why do they need encapsulation?
- What is the difference between VLAN encapsulation and VPN encapsulation?
- Which is safer:
  - black listing (specifying what is forbidden, allow the rest) or
  - white listing (specify what is allowed, deny the rest)