

Exam questions

Monday, 28 January 2019 10.40

Firewall:

Firewall beskytter mod uønsket traffic og tillader kun den rigtige traffic fra internettet.

Ideen er at beskytte data mod andre. Og beskytte ressourcer som: memory, cpu og disk. Og beskytte ens identitet

Explain internet firewalls and some of their advantages and limitations.

Fordelen: First shield mod uønsket traffic. One-way mirror der beskytter LAN mod internet probes etc. Firewall lader brugeren blockere diverse porte for at sikre brugeren bedre.

Ulempe: Falsk sikkerhed. En bruger vil stadig blive inficeret af malware fra en email eller lign. Kan bliver ret uoverskuelig at for større networks for administratoren at åbne og lukke porte korrekt for brugerne.

Describe different types of hackers

Joyriders

- Bored people som bare vil se om de kan.

Vandals

- De kommer bare for at ødelægge

Scorekeepers(script kiddies)

- Notere ned hvor mange angreb de kan lave

Spioner

- Seriøse hacker som gør det for pengene/regering.

Der er mange typer angreb som bla:

Intrusion

DoS

Information theft

Som alle kan udføres på mange forskellige måder.

Explain host security vs. network security

Host based giver brugeren en høj grad af frihed og flytbarhed, og ofte en feature rige! - MEN de er også lettere at 'snyde' fremfor network-based. Og HVIS en host bliver kompromitteret kan en angriber muligvis eskalere hans rettigheder og slå firewall fra eller installere andet malware.

Network based er en stærkere barriere og har større mulighed for at fange traffik genereret af en bagdør. Network based firewalls er også en enkelt device som gøre angrebsfladen mindre. Og selv hvis en angriber kommer igennem network based så skal han stadig angribe selve hosten så her er der øget kompleksitet.

What is a security policy ?

En sikkerhed politik er den dagsorden brugeren beslutter systemet skal følge. F.eks om firewall tillader alt slags traffik eller nægter den med det samme medmindre et explicit er statet hvordan det skal gøres:

- Deny all

- Deny all
- Accept all

Det er overvejelse man skal gøre hvor man vægter:

Confidentiality - Beskyttelse mod uautoriseret adgang

Integrity - Sikre at information ikke bliver ændret undervejs og kun af dem der må!

Availability - Hvem og hvornår har adgang og hvilken form for adgang skal de have?

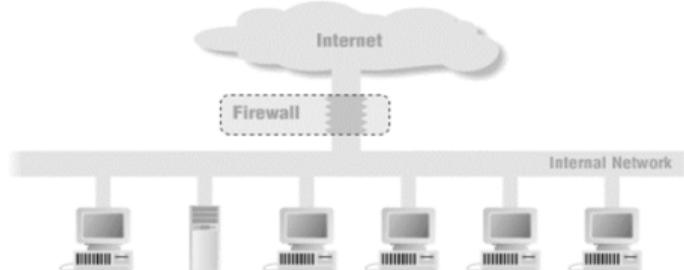
What is a dual-homed firewall ?

En type application based firewall som tilføjer et sikkerhedslag til ethvert untrusted network. Den tillader ikke direkte IP traffik mellem the trusted network(home) and untrusted network(internet).

Modsat et pakke filter firewall så blokeret dual-homed firewall for alt IP traffic mellem Internet den beskyttet side! Servicer og andet bliver så tilgået af proxy servere på gateway'en. Simpel men effektiv.

What is a screening router?

Dette er en package fitlering firewall der sidder mellem network og internet:



Flere screening routere can kombineres og skabe forskellige zoner med varierende sikkerheds niveau.

What is an Application-level gateways ?

It augments a firewall eller NAT and filters the applications data to the correct places.

The proxy-server hides the client side IP and other information. The client then connects through the proxy to any website and other network application.

Enten skal application-level gateway vide om pakkerne og hvilke port der skal åbnes for dem eller også skal pakkerne monitoreres og SÅ åbnes en port til dem. Pin-hole firewall.

What is a bastion host ?

Det er en enkelt computer der kører en proxy/firewall og er placeret mellem internet og hjemmenetværk. Det betyder også den er ret åben for angreb. På denne maskine skal der kører forskellige application der forhindrer intrusion da den er først(og muligvis eneste) forsvars værk.

Explain tarpitting

Tarpitting er når sys. Admin. Bevidst forsinker indkommen traffic for at forhindre klienten bliver 'overrendt'. F.eks med en angriber der sender masse- spams så kan sys. Admin. Sænke hastigheden og/eller fange disse emails i denne "tarpit".

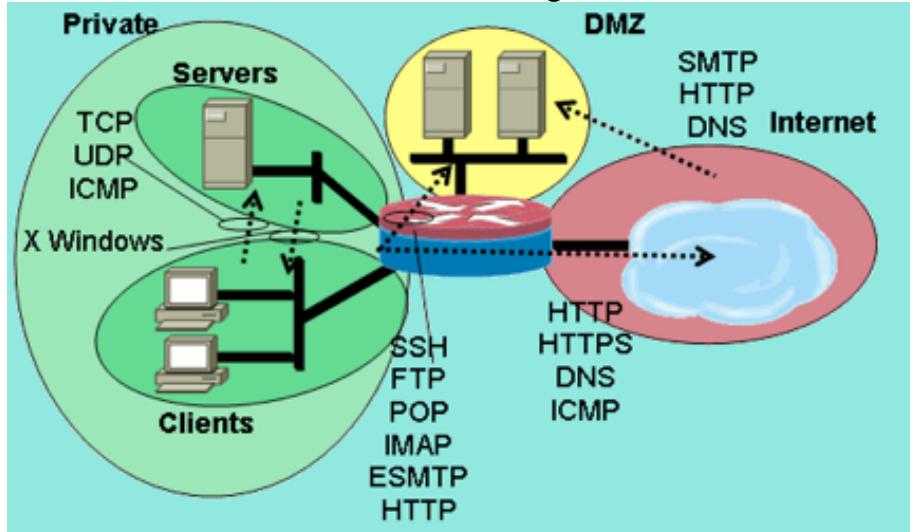
Explain zones

Kun DMZ er utsat for internettet og mulige angreb. De private zoner er indelt i forskellige zoner, som backups, VPN, hypervisor etc hvilket gør det letter at styre interfaces og hvem der må connecte med hvem.

Man skal tænke flere forskellige metoder og ikke bare flere forskellige firewall ->

Bil har både lås og ALARM.. Ikke bare mange låse.

Man er kun så stærk som sit svageste led!



What is wireshark and what can it be used for ?

- Debug network
- Debug protocols
- Write dissector in LUA
- Run on remote host (tcpdump, tshark)
- Decryption
- Understanding

What are netfilter and iptables?

IPtables er et program hvor man opstiller en masse regler for traffikken på netværkskortet så snart pakkerne bliver modtaget. Det er en måde at sørge kun tilladt traffik på de korrekte porte og IP addresser kommer er tilladt. Alt andet kan f.eks blive droppet.

Netfilter skal ses som et firewall framework til linux. Det indeholder flere forskellige moduler hvor IP tables er sådan et modul.

Explain Network Address Translation

En lookup table der oversætter den éne eksterne adresse til flere local addresser. Tænk på receptionisten der viderestiller en kunde til dig på kontoret et sted.

Explain Dynamic Port NAT

Dynamic Port NAT er når den interne adresse vil sende en pakke til internettet, så bliver addressen og porten ændret til den første ledige der er i tabellen og når der kommer return svar fra det eksterne (internet) kigges der i tabellen og pakken routes til den korrekte interne IP adresse og port.

Explain Static Port NAT

Det samme som dynamisk udoger at tabellen er "hardcoded" forudbestemt og ikke ændrer sig i.e. statiske eksterne IP og port.

Explain DDoS attacks

DDoS handler om at gøre offerets service/maskine ubrugeligt og kan opnås ved en lang række forskellige metoder hvor de mest almindelige er at flooding. Det kan være pings/SYN eller ARP beskeder. Eller andet.

Forside tilbageskrivning

Explain the SYN attack

A SYN attack is a type of denial-of-service (DoS) attack in which an attacker utilizes the communication protocol of the Internet, TCP/IP, to bombard a target system with SYN requests in an attempt to overwhelm connection queues and force a system to become unresponsive to legitimate requests.

(tcp/ip er når 2 computere prøver at forhandle om communication såsom ssh, http osv. Også kendt som TCP/IP 3-way handshake)

Explain the idea of policy based routing

Ideen er man router pakker baseret på politikker og ikke dest. Eller source addreser. Det kan være

Explain SYN cookies

En metode til at undgå SYN attack hvor SYN queuen lader som om det bare bliver større. Serveren dropper at smide mere i SYN-queuen men svarer stadig med SYN-ACK til dem der allerede er queued.

Cryptography:

Explain the pros and cons of using TLS

Transport layer security. - Cons

Latency - Was a problem years ago but not anymore.

<https://www.ssl.com/article/pros-and-cons-of-ssl-https-tls/>

Cost of certificate - price varies but then consider the level of security one gains!

Mixed modes - if implementation is setup wrong so server accepts http and https to some data visitor might get confused and get warning messages.

Its a protocol NOT a cipher.

- Pros

Trust - shows in the browser that the site is secure.

Verification - Guarantees your visitors that you are who you say you are.

Integrity of Data - Garantere data integrity of data. Without SSL one can intercept and change data to and from the web server.

Google - Are ranking searches whether or not a server uses SSL.

Explain symmetric encryption

Bygger på shared secret ide. - Vi kender begge den hemmelige nøgle som bruges til encryp/decrypt.

Explain asymmetric encryption

I dette eksempel bruges et nøgle par, public og private. Hvis jeg sender noget tager jeg modtagerens public key og encrypter med hvilket betyder kun modtagerens master key kan decrypt beskeden.

What does the ciphersuite specify?

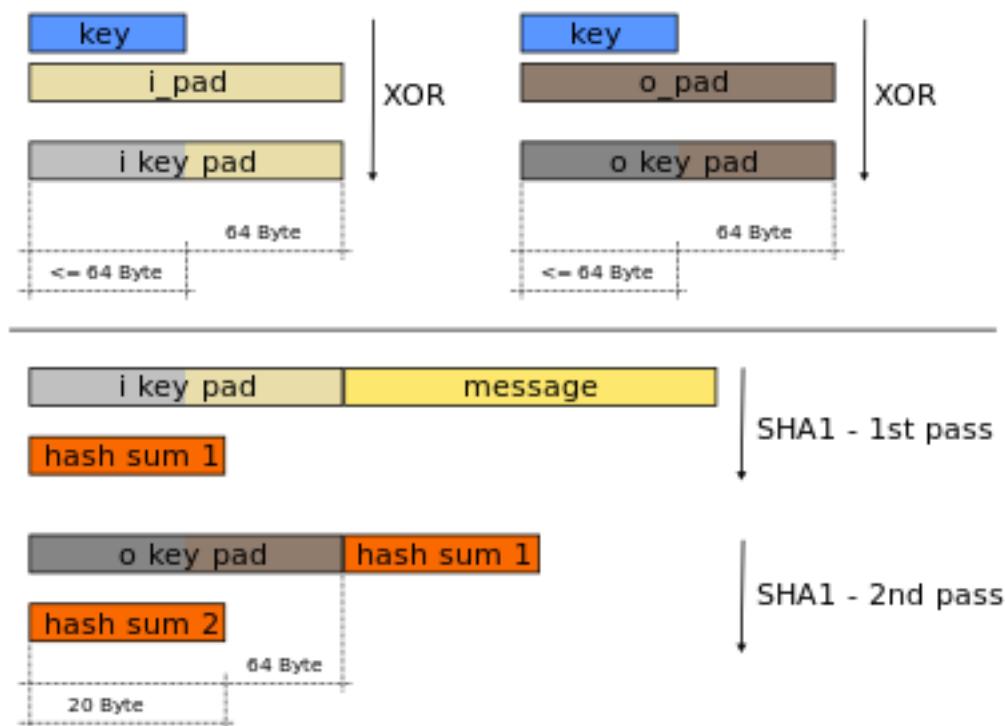
It specifies:

- Key exchange algorithm
- Encryption algorithm
- Message Authentication Code (MAC) algorithm
- Authentication algorithm

Explain HMAC

Hash-based message authentication codes - En hashing functions koblet med en hemmelig nøgle som bruges til både at verificeres data integriteten og

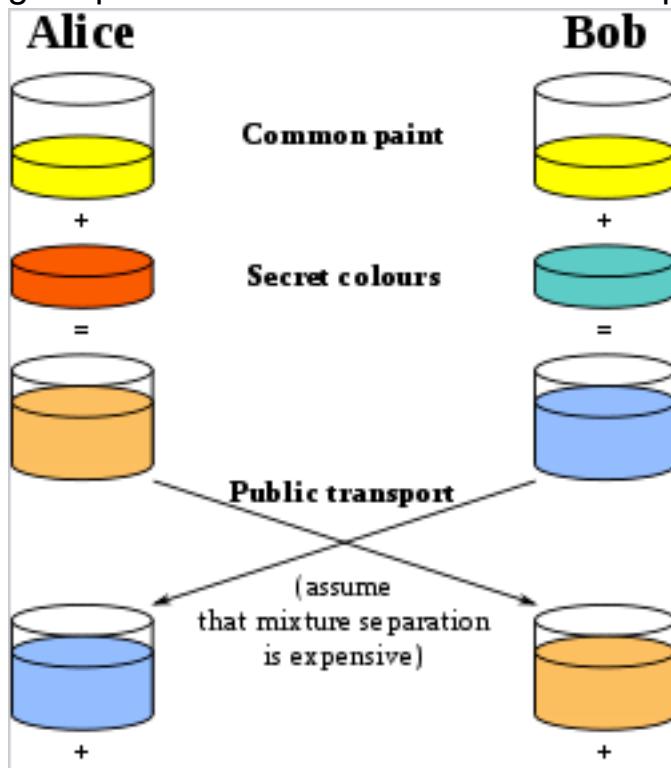
authenticity(afsender) af en besked.



Nøglen bliver brugt 2 gange med forskellige pads. Hernæst bliver vores besked og padded nøgle hashed én gang. Og så bliver hash_sum1 og padded nøgle2 hashet igen!

Explain Diffie Hellman (without math)

Alice og bob bliver enige om en fælles farve(et stort tal) og vælger nu en hemmelig farve(stort tal) og får den mixede farve. Det mixede farv sendes nu over til dem hver især og de tilføjer deres hemmelige farve(tal) og får så en delt hemmelighed. Hvis nogen lytter med når de deler deres blandede farve skal så gøre processen omvendt hvilket er computationelt tungt!

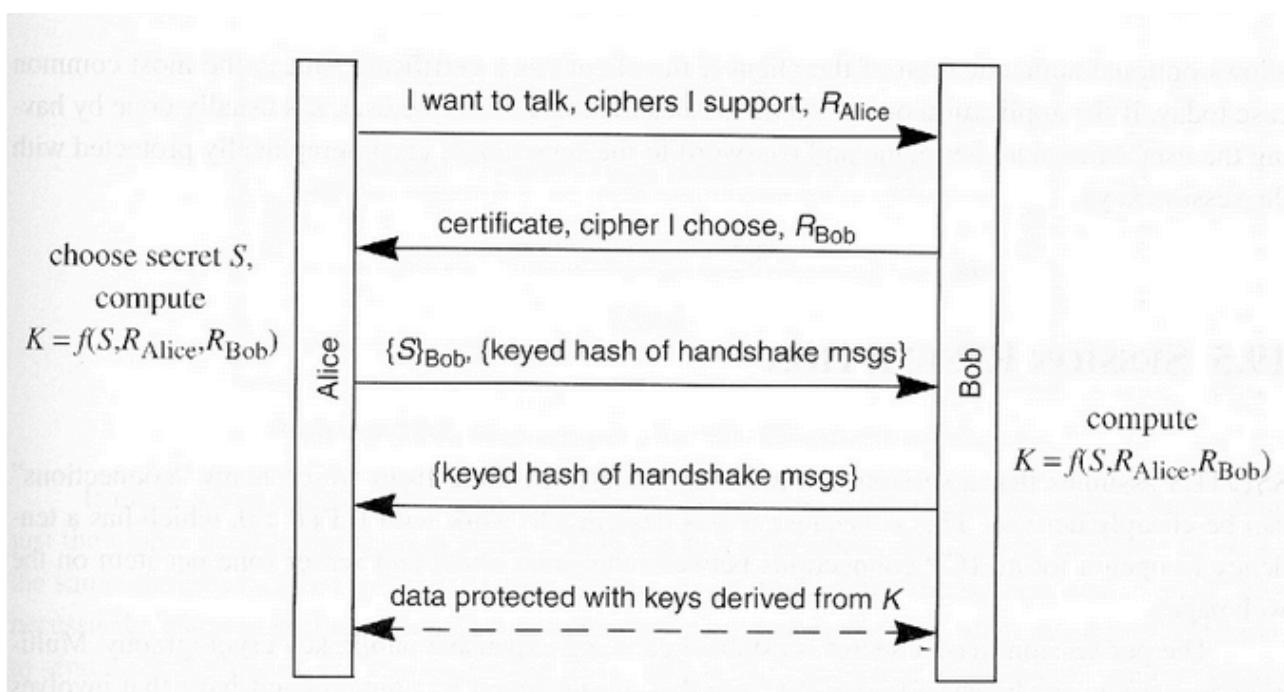




What is the Pre-Master key and Master Key and what is their purpose ?

Pre-master key er i besked 3, fra Alice til bob hvor den fælles secret, chipers fra Alice samt certifikat med valgt cipher fra bob.

Master key er den fælles nøgle de nu bruger for at tale sammen encrypteret!
Pre-master key bliver lavet med "random" tal fra client og server og encrypterer den med serverens public key. Nu bruger både server og client den encryption til at lave en master key. Pre-master key er også for at få alle de forskellige ciphers til samme format!!



Describe the essential values in a X.509 certificate

Det er en standard indenfor certificater som bruges nærmest overalt. Den indeholder certifikatet og underemner. Certificates signatur og signatur algoritm. Underemner:

- Version Number
- Serial Number
- Signature Algorithm ID
- Issuer Name
- Validity period
 1. Not Before
 2. Not After
- Subject name
- Subject Public Key Info
 1. Public Key Algorithm
 2. Subject Public Key

What is signing and what is it used for ?

Signing er at den når en side/certifikat eller andet encrypter med deres private

Signing er at den har en side/certifikat eller andet encryptet med deres private key således at alle med denne persons public key kan decrypt, derved ved vi alle at den person/firma/side ER den han siger han er!

What is a Certificate Signing Request(CSR) ?

Det er en besked en ansøger sender et certificeres firma for at få certificeret ansøgeren digitale ID. Denne besked indeholder forskellige information om ansøgeren.

Information
Distinguished Name (DN)
Business name / Organization
Department Name / Organizational Unit
Town/City
Province, Region, County or State
Country
An email address

What is a Certificate Authority(CA)?

Det er den autoritet der underskriver andre certificater. Det er dem vi alle stoler på! - det er en måde at kontrollere om vi kan stole på det certificat som en server sender ud.

Explain chain of trust

Chain-of-trust kommer sig af der er nogle få SELF-SIGNING certificater som vi alle har besluttet vi stoler på! - det skriver så ud og stoler på nogle certificater som har lov til at udskrive certificater til andre. Grundet den måde certificaterne er bygget op på kan vi altid se hvem der er root, og om det er nogen vi stoler på!

Explain how the TLS client verifies the authenticity of the TLS server

The client verifies server ved at kigge på serveres certificat og denne chain of trust. Dette vil afsløre om det er et certificat man tør stole på!

Explain the MITM attack

Det er muligt at lave et MITM attack men kun så længe at clienten stoler på det "onde" certificat. Det betyder at man skal have listet det "onde" certificat in på client maskinen først! Efter det er det muligt at encrypt/decrypt information on the fly.

Hardware Hacking

What are the main attack surfaces of a computer?

I he tysiske brugertlade på computeren - Fysisk alle dens inputs (USB, ethernet etc.)

Network

Software

What is a keylogger and how can you protect against it?

Keylogger - could be a USB dongle one put between the keyboard and computer.

You can protect against it by locking your computer away.

Keylogger er et program der logger hvilke knapper, clicks eller lignende der bliver tastet.

Hvis keyloggeren sender data over nettet ville en stærk firewall måske bremse data'en ud.

Malware scanner, Kig efter om nogle underlige programmer logger input fra keyboarded.

Begræns fysisk adgang til computeren.

What is a virtual user device and how to protect against it?

How to protect against a virtual device - Rubber ducky works as some device (keyboard f.x.) and starts to do all the stuff af user can do. Virtualized through the hardware, USB dongle.

Virtual device er devices som raspberry pi zero eller USB som emulere andre devices, som et keyboard for så at launche deres malware hos targeted host.
Begræns adgang til fysisk hardware.

Aldrig åbne ukendt USB.

Åben USB container/closed/contained enviroment.

What is a network tap and how can you protect against it?

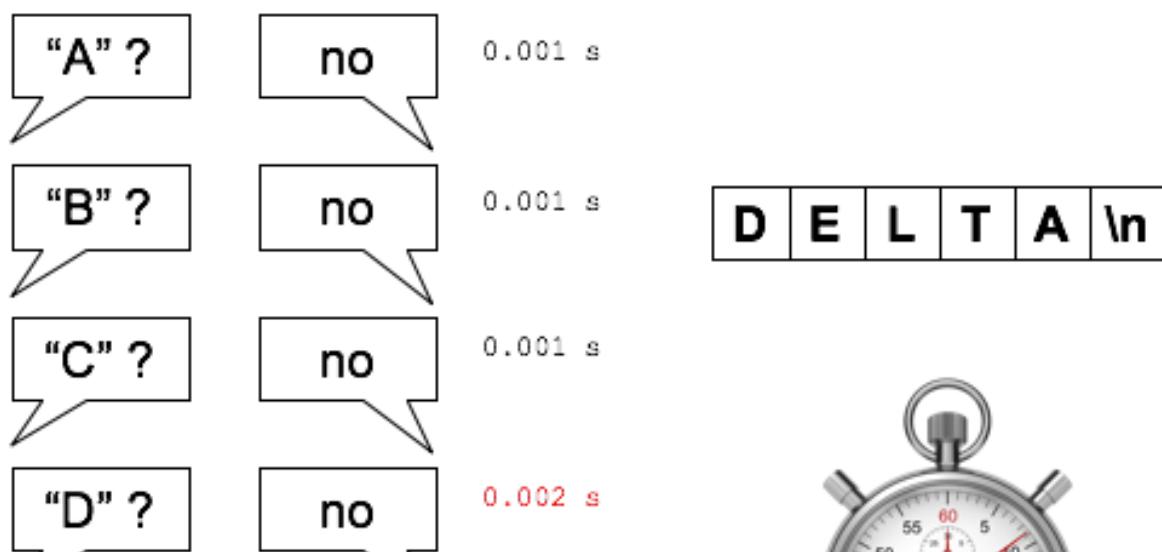
A little device to put inbetween two points in a network which sends the data unchanged to a third party thus being undetectable.

Limit hardware access!

What is a network tap and how to protect against it. - Encrypt traffic. hide wires etc.

Explain the timing side-channel attack for a simple password.

Vi "brute force" spørger om password og timer "response" tiden får at få en bedre idé om hvad passwordet kunne være.





Explain the reasons for hacking hardware

Udvidet funktionalitet/extra features

Modify hardware

Få adgang til hemmeligheder

Kompromitter en device el. tjeneste.

What is disassembly and how can we protect against it?

Takes a binary code and makes it to somewhat readable assembly code. Again encryption.

Disassembly is software that looks at e.g. CPU images and finds out how it works and make it easier for the user so that one can modify or extract hidden data.

It shows code so that the user gets a quick look at what it does.

One can do something about it by encrypting the firmware and other code.

Explain virtual memory

Virtual memory is an expansion of RAM so that there is enough for all the things a user wants. Therefore it checks RAM areas that have not been used for a while and copies them to the hard drive. This is similar to having infinite RAM due to the fact that HDD is much cheaper than RAM.

The trick is to have not enough RAM to run everything simultaneously and so the user will notice it is slow in virtual memory compared to task switching. If one is dependent on virtual memory then one will notice a performance drop.

Explain the anatomy of a program in memory

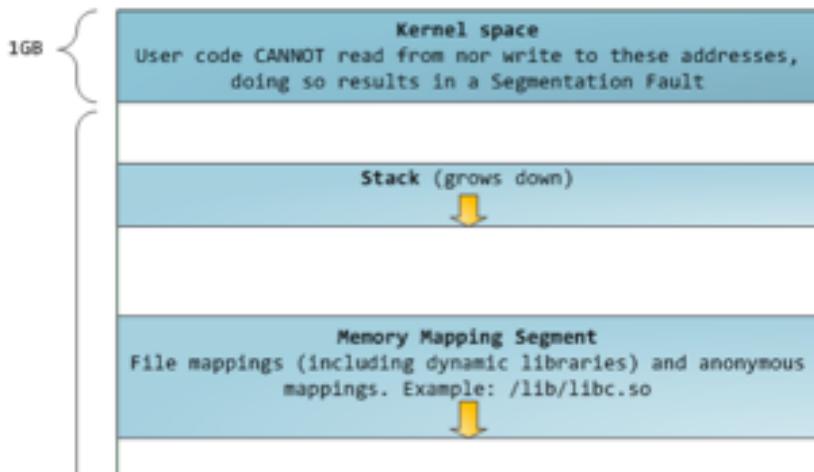
Non-static code that needs to be executed is located in stack's memory

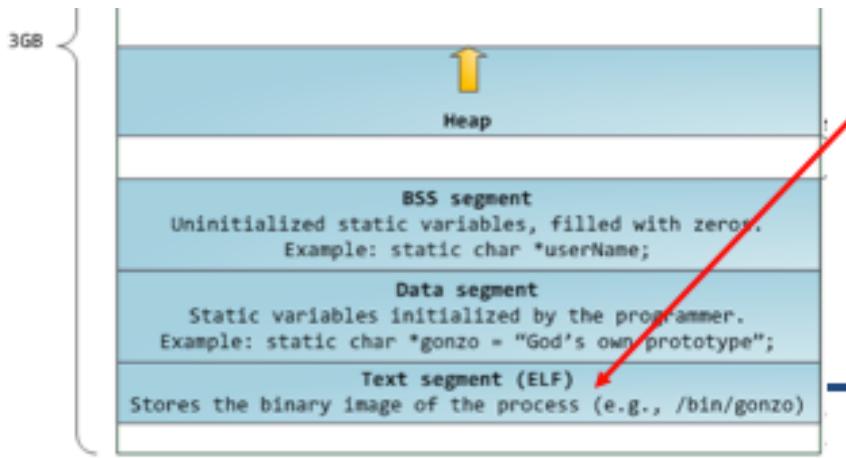
Ptr = newObject(); - This is a "random" /new objects, pointers to other objects are located in heap's memory

Uninitialized static variables are located in BSS segment.

Data segment contains static and initialized variables.

Text segment contains strings.





Explain fault attacks

Provoke faults by:

- Heat
- Radiation
- unintended use
- Laser
- Electro-magnetic impulses
- UV lights
- ...
- Stuck-at faults - DDoS. Annoying. Don't really do much other.
- Bit-flip faults - typically in memory
- Set/reset faults -
- Random faults - break a statemachine by misusing the statemachine and get the statemachine into a "illegal" state. Spillemaskiner is especially a machine where people wanted to break the statemachine to cash out
 - Transient
 - Permanent
 - Destructive
 - Precise control
 - Loose control
 - No control

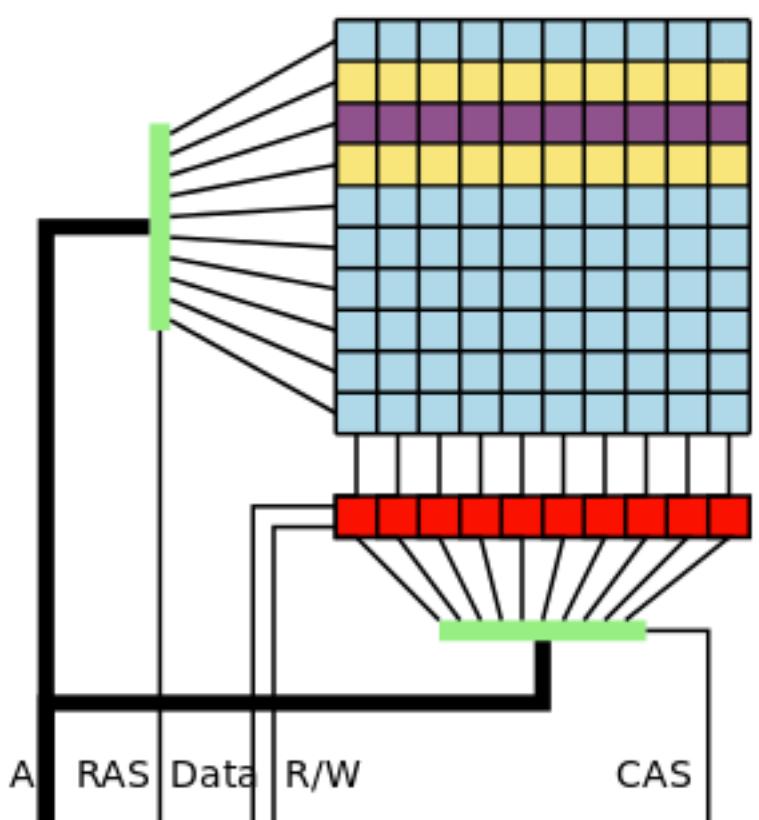
Explain rowhammer

Udnytter fejl i DRAM ved at inducere en capacitor fejl i DRAM som så gør at der bliver peget et forkert sted i HDD'en hvor vores ondsindede program ligger. Hvis man så ligger sin ondsindede kode i et program som Ping som har sudo rettigheder kan man nu køre det onde program som sudo!

Ideen er man hamrer /skifter mellem 2 rækker i DRAM for at inducere strøm i nogle andre rækker og få en 'lucky bit switch' og derved pege at andet sted i memory.

DRAM rækker bliver aktiveret over hele linjen og kolonnerne vælges så får at skrive i ram. Dette lades op i en capacitor man da den mister strøm over en

periode skal den have strøm igen aka genskrive data'en(bits) Dette hedder (cache) refreshing.



Double side rowhammer. Hamrer begge sider af lilla række for at inducere en fejl. Ideen er vi fylder hukommelsen med alt muligt evt beder om mere og så håber vi at nogle af de permissions der ligger i det program vi angriber med ligger mellem de rækker vi har adgang til, og så hamrer vi derud og håber på et bit flip der giver os udvidet tilladelser.

Forsvar evt mem ecc(error correction) RAM der fungerer om en check om nogle bit er blevet flippet. De er ikke så effektive som de giver sig ud for.

Man kan holde øje med alle tilgange til hukommelse fra systemet, men det er ganske dyrt at monitor alt.

Man kan øge refresh rate men så bruger man mere power, evt dræner en telefon hvis man bruger den der.

En ny ide er at øget refresh rate på de rækker der er i brug. Så selv om du hammerer på nogen så refresher vi bare desto mere de andre.

basically a attack where you attack the physical ware. 2 years old approx since google announced it. Still an active attack. People can get out of containers and acces other people containers. Cellphones are suseptible to this attack!

The basic of rowhammer is explain by first understanding the DRAM cell and how it works!

Cache memory will be asked from the CPU "do you have this in memory" to level 1 cache. If it does not then it will ask level 2 cache. If it does not have it either it will fetch it in the DRAM and put it up to level 2 cache and then level 1 and lastly the CPU.

The central part of this attack is that you "hammer" two rows and flush the cache. We do that 1.2M times per 64ms refresh of the ram.

We need to ask the page table of where the memory is located. We do not have write rights to the page table. However we have write access to the physical memory.

This means we "spray" the physical memory with pagetables and then we start hammering and we want to induce a bitflip. When we get the bit flip the pagetable will point to something else!

And then we can write our own pagetable with read/write rights i.e. access to entire memory.

Then we find a SUID program (i.e. ping) which has sudo privileges!

Overwrite that program with your own code.

NOW you can run your own code as root.

PROTECTION against it.

Larger capacitors. - needs more charge to read the RAM.

Shorter refresh periods. - instead of 64ms then go to 32ms but then you pay with power consumption and access.

Disable the 'clflush' - done by Google and Amazon. It is implemented in the assembly structure so that is interesting.

Compile a kernel that does not support ping and other sudo programs not needed when deploying.

Explain meltdown

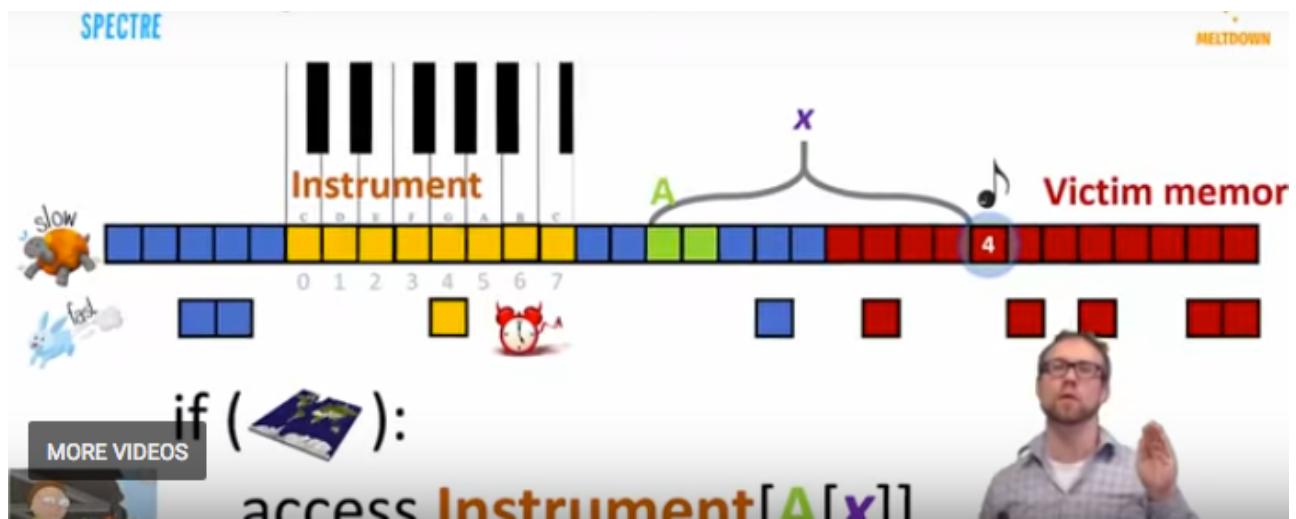
Meltdown bygger på hvordan et program eksekveres i CPU. RAM er langt om så derfor lægges noget af data'en over i en cache!

Når et program køres så bliver nogle af linjerne eksekveret selvom der er en if statement fordi at ellers står CPU'en bare og venter. Det hedder speculative execution. Man skal passe på med at ændre for meget for hvis CPU'en spekulerede forkert skald en rulle det der blev gjort tilbage igen.

Meltdown udnytter dette speculative execution.

Vi har noget memory vi kontrollere men vi vil læse noget vi ikke må læse og vi opretter et instrument. Først opstiller vi en absurd if statement for at få den speculative execution i spil der læser på forhånd fremad. Og det farlige er så at svaret - det vi ikke må læse (x) bliver lagt i cachen! - det gør så at vi kan aflæse cache med et side-channel (timing) attack som vi gør med svage passwords!

Meltdown of spectre er næsten ens men spectre tillader kun fra samme program. En fane til en anden fane hvorimod meltdown er hukommelse fra et program til et andet.

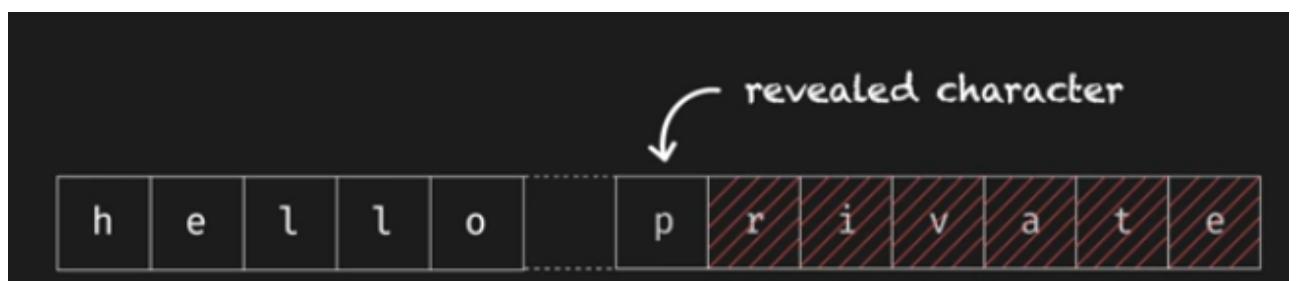


Igen:

Fordi CPU ikke kører kode segmentielt men derimod -> out-of-order(faster) kan man tilgå privat hukommelse. Hvis man prøver at læse privat hukommelse vil CPU'en fange det og smide en segmentation fault og stoppe programmet men desværre er det man vil læse allerede blevet lagt over i cache'en for at tilgå det hurtigt. Det angriberen nu skal gøre er at gætte value of the secret. -> side-channel attack!!

Så laver man et array med alle værdierne i.e characters=[a,...z] og prøver så at læse characters[secret + 1] og timer det. Når et af dem giver et meget hurtigere svar ved vi at det "bogstav" allerede var i cache'en!

```
secret = readCharacter(1000); Share  
  
characters = ['A', 'B', 'C', ... 'Z'];  
  
characters[secret]; 60ms  
characters[secret + 1]; 60ms  
characters[secret + 15]; 3ms  
  
MORE VIDEOS
```



Hvis man kan overskrive return addreser med 'crafted string'.

**How can you inject arbitrary code?
(We do not need all the gory details,
but the idea of what you have to do)**

Hvis vi ved hvor koden er kan vi overskrive den retur adresse så når funktionen returnere så executer den vores onde program.

What has to be known to get the code executed?

Retur addressen eller sådan ca der hvor subroutinen køres.

What can be guessed?

Vi kan prøve at gætte på hvor hvor retur addressen er da stacken ikke er sådan særlig stor. 100 til 1000 linjer pr. subroutine.

Vi kan også indsætte en masse NOP's før vores onde kode får at øge chancen. Det betyder også at hvis bare en NOP bliver ramt så eksekveres vores angreb.

Which counter measures for stack overflow attacks exist?

Bruge "stærke" sprog hvor buffer overflows opdaget såsom C# og java-
Brug sikre biblioteker som ikke har tendens til overflow som strncpy kan!

Brug stackshield som gemmer væk return addressen og selv hvis addressen er ændret i den "normale" stack så bliver den "gemte" værdi brugt !

Stack guards: Hvis man vil overskrive return addressen skal man som ofte overskrive en masse FØR return addressen. Ved at smide en kanariefugl ind i koden(hemmeligt ord/tal) som man så tjekker på i run time og hvis det ikke er det skal være ved man at nogen har forsøgt sig at gøre ting ved stack'en.

What can you do on programming level?

Test koden og sig højt hvis du finder fejl! - aldrig bare ignorerer det.
Ingå at bruge pointere og pointer arithmetic.
Harden your code against unreasonable input.
Make sure, it survives outright hostile input!

How can you spot/test for faulty code?

?????

Which operating system measures exist?

Randomization af start addressen på stack'en! (Linux kan gøre det)
Hvis man ikke ved hvor den starter kan man ikke gætte den :D

Linux can also make the stack NON-executable. Dvs at "ond" kode ikke kan executes selvom det skulle ligge på stack'en

Which hardware measures exist?

How can the compiler protect

(i.e., changes of the program to harden against attacks)?

Array bounds checker - mitigeere overflow problemer

.....,

Kode sprog der er sikrerer. Fejlene stammer fra C.

Bruge "stærke" sprog hvor buffer overflows bliver opdaget såsom C# og java-

Brug sikre biblioteker som ikke har tendens til overflow som strncpy kan!

Brug stackshield som gemmer væk return addressen og selv hvis addressen er ændret i den "normale" stack så bliver den "gemte" værdi brugt !

Stack guards: Hvis man vil overskrive return addressen skal man som ofte overskrive en masse FØR return addressen. Ved at smide en kanariefugl ind i koden(hemmeligt ord/tal) som man så tjekker på i run time og hvis det ikke er det det skal være ved man at nogen har forsøgt sig at gøre ting ved stack'en.

Are there (counter-)counter-counter measures?

Shadow stack til at sammenligne med!

Seperer data og control flow i forskellige stacks.

Connection to attack surface?

??

general attack scenarios?

Steal, spy exalate priviliges, destroy

what is the aim of stack overflow based attacks?

Steal, spy exalate priviliges.

Could be passwords could be all sort. Hvis du finder ind til en system admin er det ikke til at sige hvad man kunne finde.

Det kan være at nedlægge en service

Fun question:

Which problems may the compiler create for the data security aware programmer?

Husk altid at slette hvad du har liggende i RAM og evt sensitiv information!

Det er ikke nok at sige til compiler at den bare skal "frigøre" hukommelsen. Hvis du vil være sikker så overskriver du hukommelses pladsen med noget andet!!