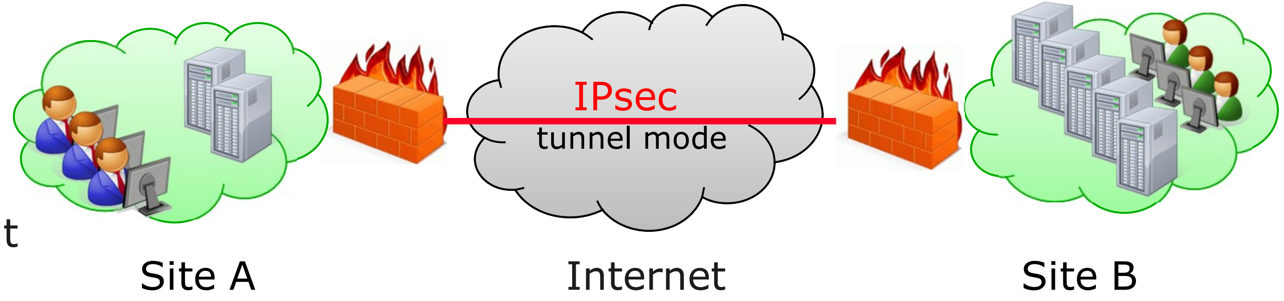
VPN systems and Network Design

VPNs: Virtual Private Networks

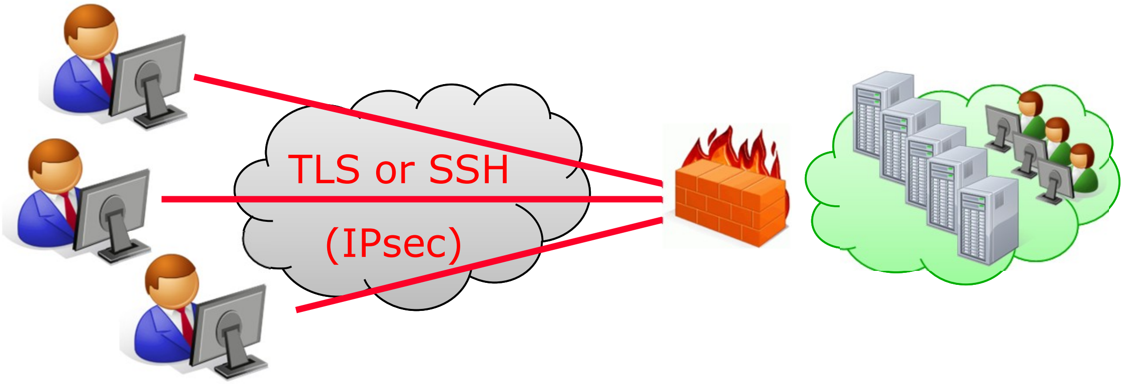
Site to site VPN

Transparent access



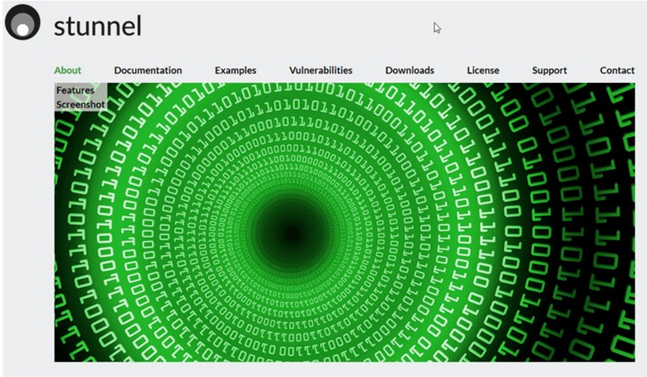
User to site VPN

Remote access for users



Application level VPN systems

- IPsec often blocked by firewalls
  - E.g. at some hotels, airports, corporate border firewalls
  - Users needs to be local administrators to configure it
- TLS and SSH can be good alternatives
  - TLS perfect for web-based access, built-in to all web browsers
  - ”stunnel” implements SSH-like tunnels for TLS (OpenSSL)
  - More functionality is possible with browser add-ons (Java, …)
- TLS and SSH have limitations
  - They are not as transparent as IPsec
  - SSH and TLS with stunnel have limitations for what can be tunneled
  - In some situations, this can be a security advantage
- Special software can offer full IP traffic tunneling like IPsec
  - Virtual network cards allow transparent tunneling of traffic with TLS or SSH
  - Drawback: more intrusive client, users need administrative privileges to install
  - Advantage: client software may be able to control what applications are allowed to communiate through the tunnel
- Takeaway: Additional software can give SSH and TLS similar functionality as IPsec



https://www.stunnel.org/

Stunnel uses the **OpenSSL library** for cryptography, so it supports whatever cryptographic algorithms are compiled into the library. It can benefit from the **FIPS 140-2 validation** of the OpenSSL FIPS Provider

Tunneling all IP traffic with SSH-VPN

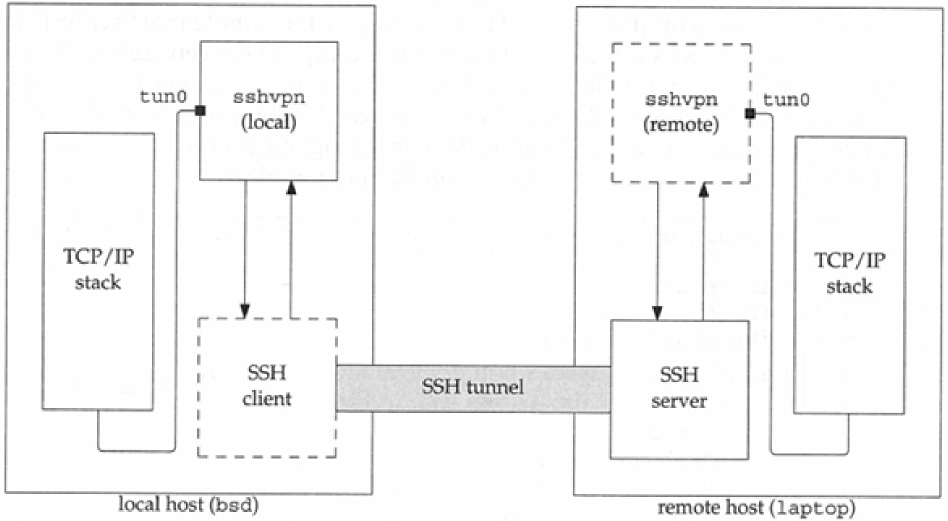


Figure 7.58 Architecture for the SSH VPN

Functionality is similar to IPsec in tunnel mode.

tun0 is a virtual network card (virtual Ethernet interface) which intercepts network packets and sends them to SSH to be securely tunneled.

tun0 announces a better route to the remote network than the physical network card.

Ref: VPNs Illustrated: Tunnels, VPNs, and IPsec: Jon C. Snader

FIPS 140-2 Approval

FIPS = Federal Information Processing Standard

Federal agencies are required to use cryptographic algorithms that are NIST-approved and contained in FIPS-validated modules. The FIPS 140-2[7]specification defines how cryptographic modules will be validated.

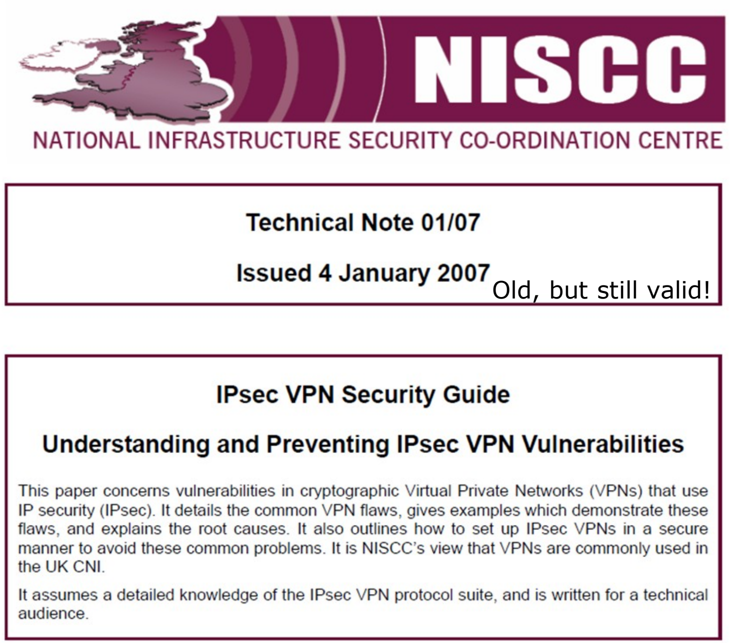
One requirement of FIPS 140 is that the module be capable of operating in a mode where all algorithms are NIST-approved

NIST-approved algorithms are specified in a FIPS (e.g., FIPS 180, *Secure Hash Standard*[8]) or in a NIST Recommendation

Because VPN technology employs a number of cryptographic algorithms, Federal agencies must be aware of whether their chosen VPN technology is FIPS-compliant now and whether it is expected to be FIPS-compliant during the entire expected lifetime of the system

FIPS 140-3 is an incremental standard which includes advancements from ISO standards (which build on 140-2)

From NIST Special Publication 800-113



NISCC

NATIONAL INFRASTRUCTURE SECURITY CO-ORDINATION CENTRE

Technical Note 01/07

Issued 4 January 2007

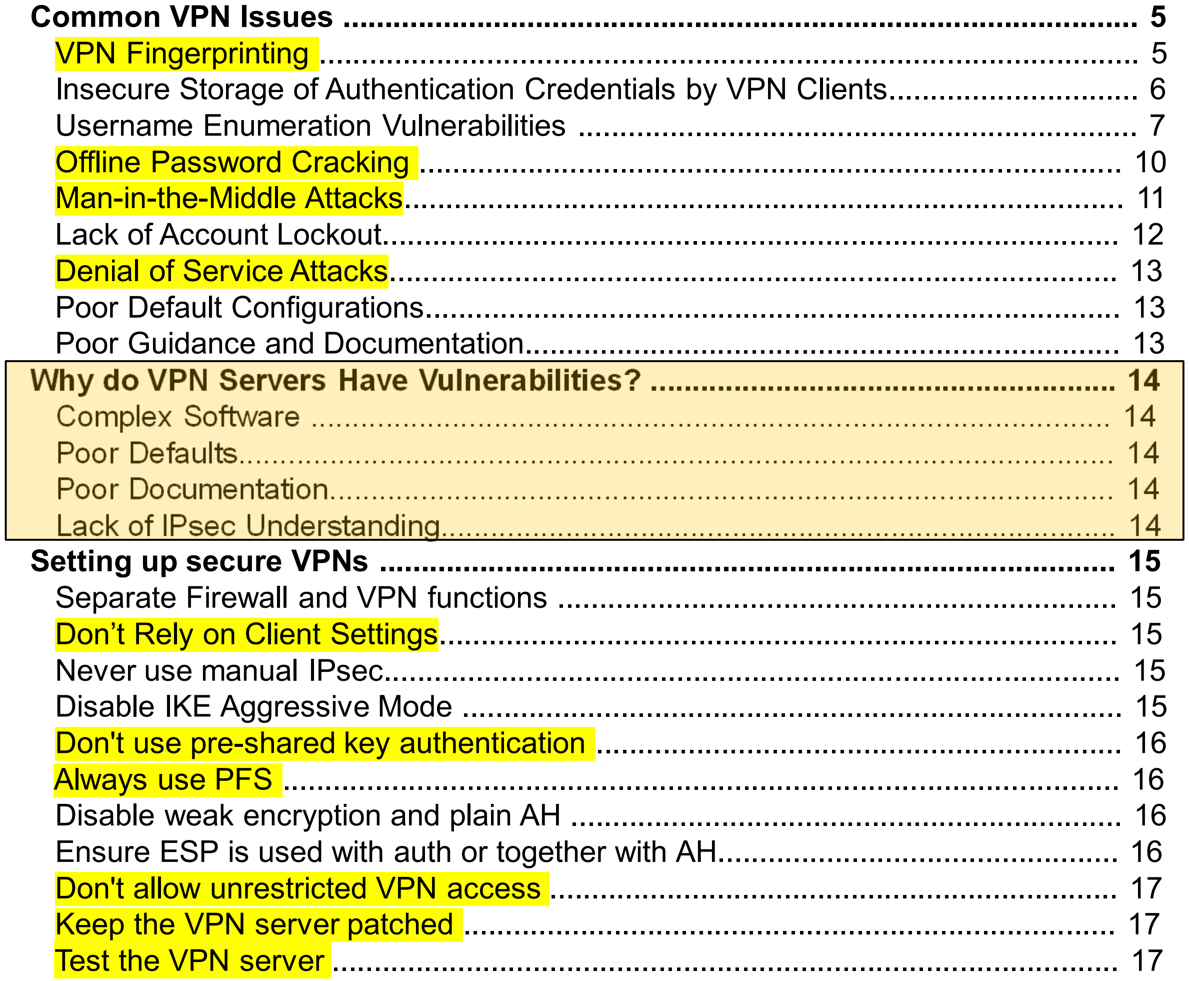
Old, but still valid!

IPsec VPN Security Guide

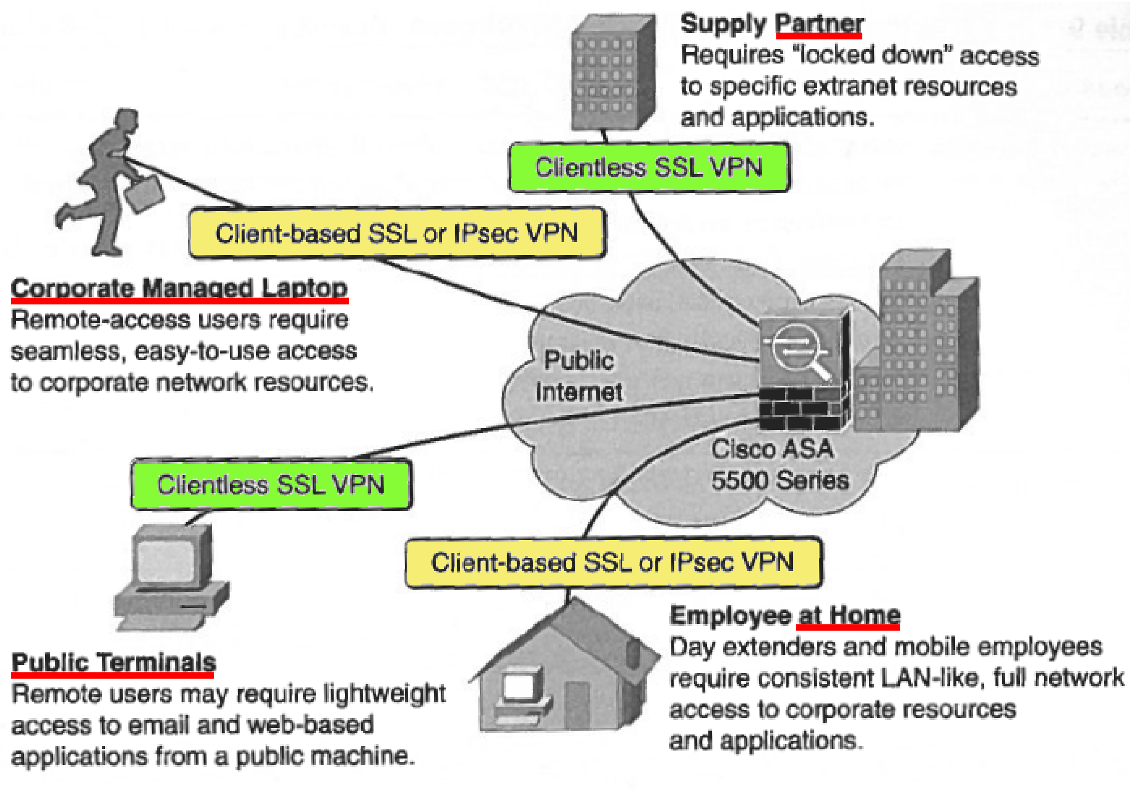
Understanding and Preventing IPsec VPN Vulnerabilities

This paper concerns vulnerabilities in cryptographic Virtual Private Networks (VPNs) that use IP security (IPsec). It details the common VPN flaws, gives examples which demonstrate these flaws, and explains the root causes. It also outlines how to set up IPsec VPNs in a secure manner to avoid these common problems. It is NISCC's view that VPNs are commonly used in the UK CNI.

It assumes a detailed knowledge of the IPsec VPN protocol suite, and is written for a technical audience.



Cisco's deployment advice: seamless vs. clientless



Supply Partner
Requires "locked down" access to specific extranet resources and applications.

Corporate Managed Laptop
Remote-access users require seamless, easy-to-use access to corporate network resources.

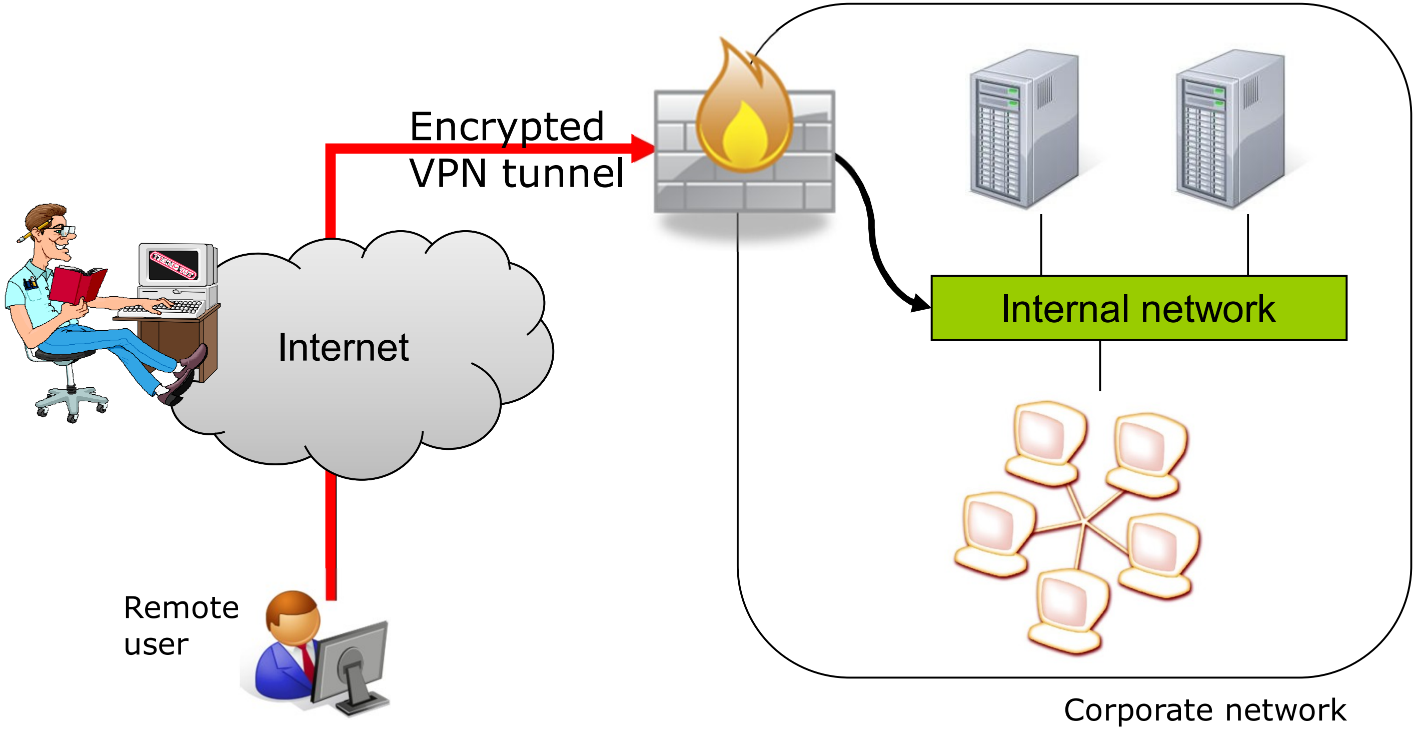
Public Terminals
Remote users may require lightweight access to email and web-based applications from a public machine.

Employee at Home
Day extenders and mobile employees require consistent LAN-like, full network access to corporate resources and applications.

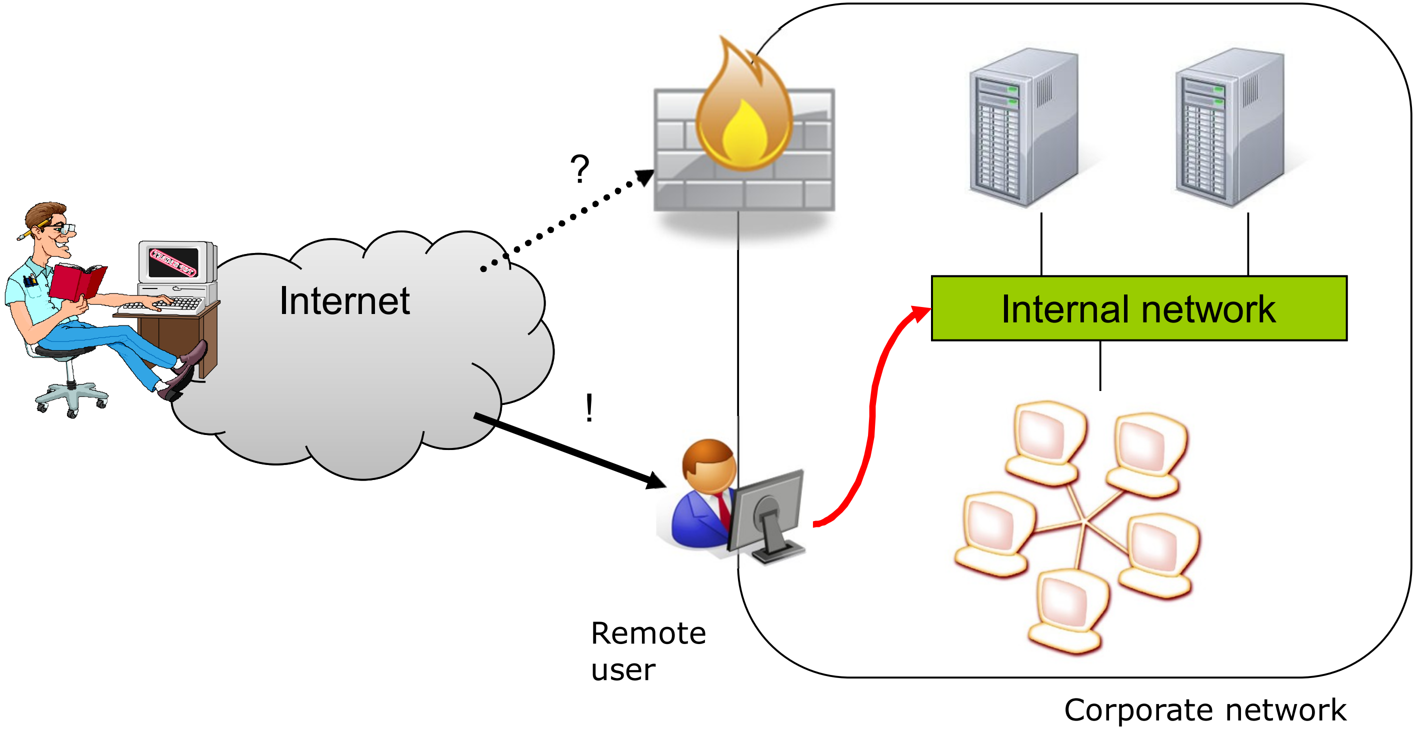
Figure 9-12 *SSL VPN Deployment Options*

Picture from Network Security first-step, Cisco Press

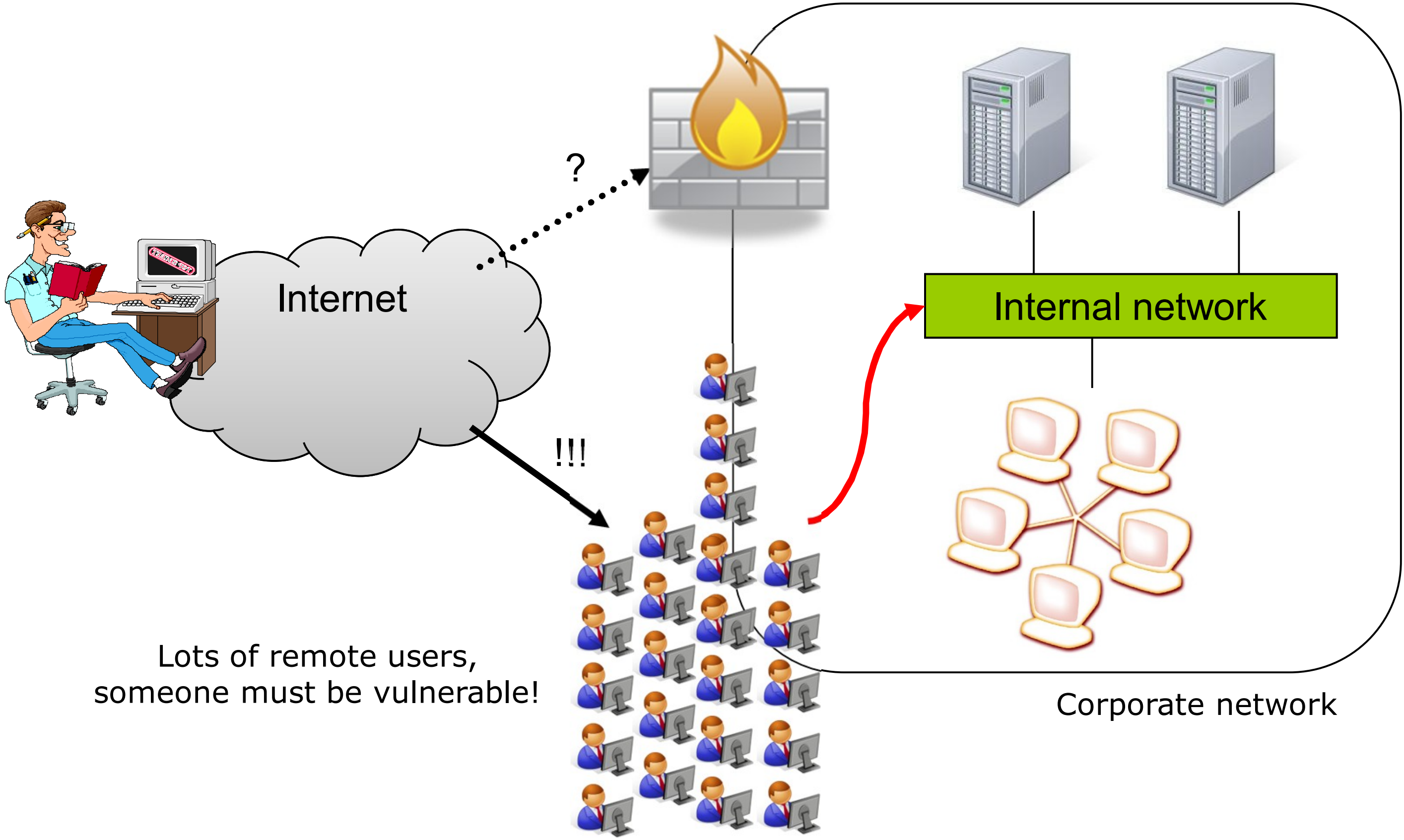
Remote access is problematic



This is the same picture…

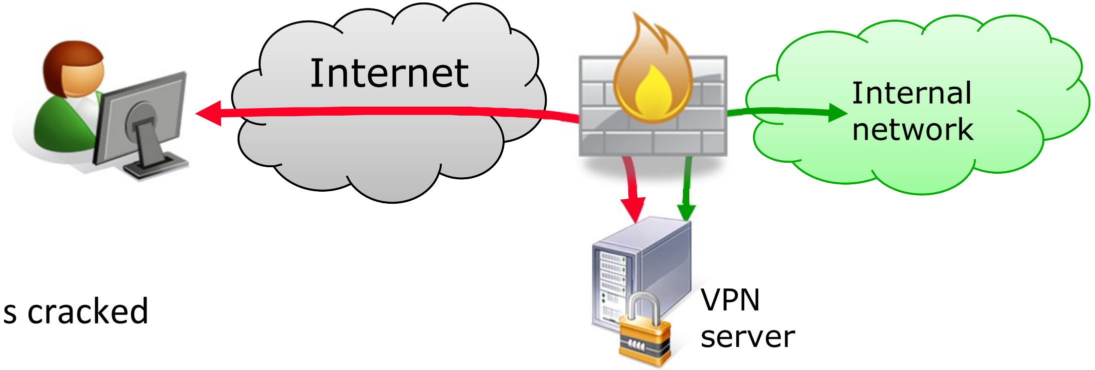


And there can be lots of targets!



Lots of remote users,
someone must be vulnerable!

Placing the VPN server



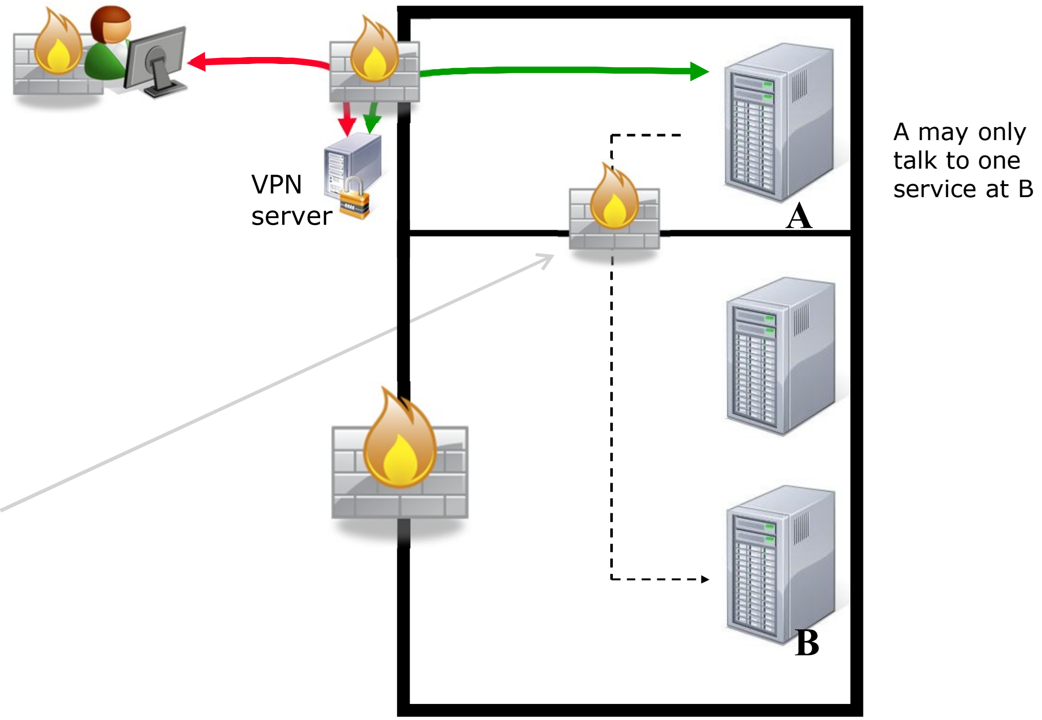
- This is a common configuration
  - Secure tunnel from user to VPN server
  - VPN server protected by the firewall
  - VPN server is not part of the firewall in case it is cracked
- The firewall can inspect decrypted traffic
- User authentication and authorization done by VPN server
- The VPN server may be located on the DMZ
  - But a cracked server on the DMZ can be used to sniff cleartext VPN traffic to the inside
  - Make sure traffic between servers on the DMZ is separated!

Internal segmentation makes it better!

- Protect external users with personal firewalls
  - Preferably centrally administered
  - Users are no good administrators

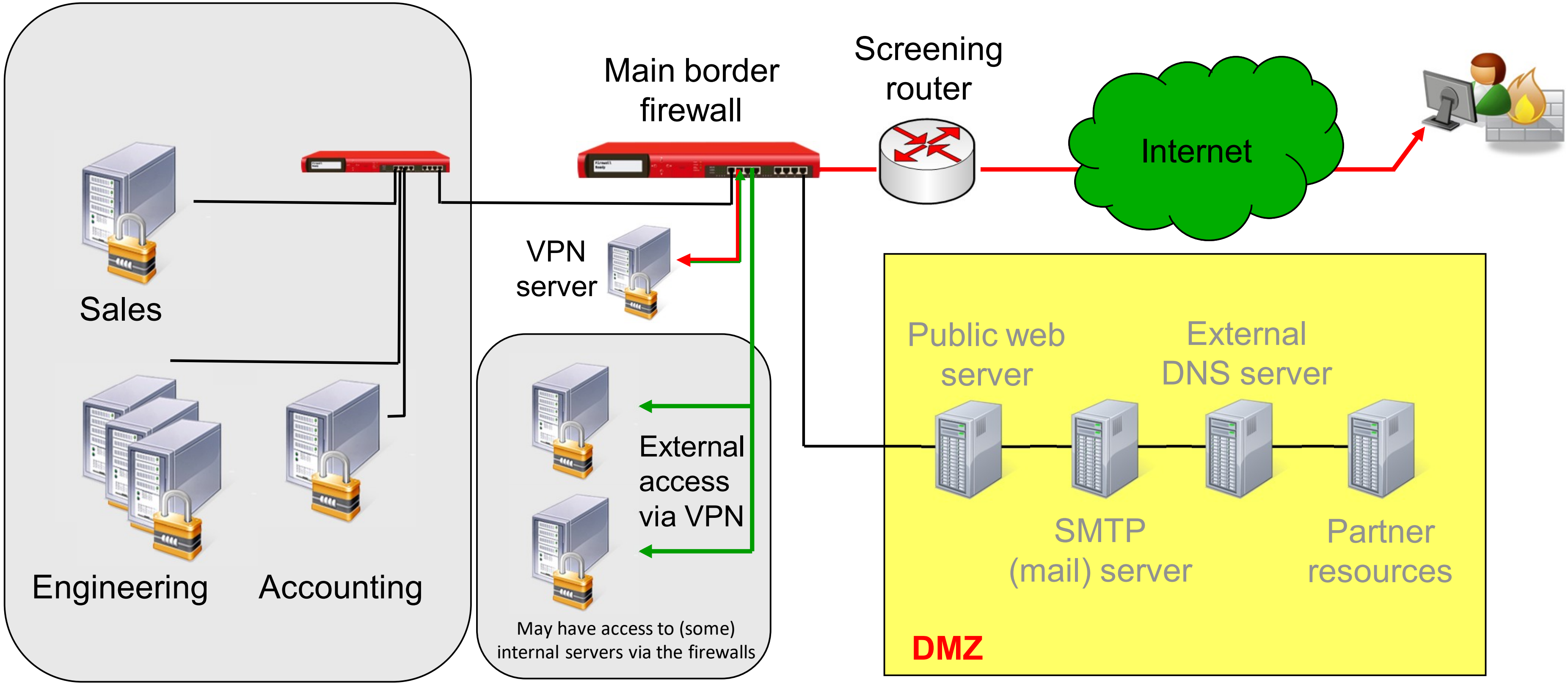
- External users should only be allowed access to as few internal systems as possible (A)
  - Never full network access!

- Limit what externally accessible servers can do
  - Place server A on a dedicated internal network
  - An internal firewall triggers an alarm if any other services than expected are accessed by A
  - Server A should also have a host-based firewall with alarm functionality



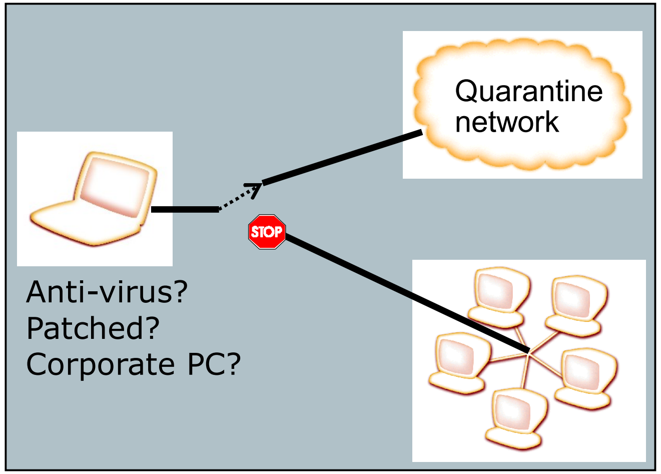
A may only talk to one service at B

Internal segmentation, similar solution

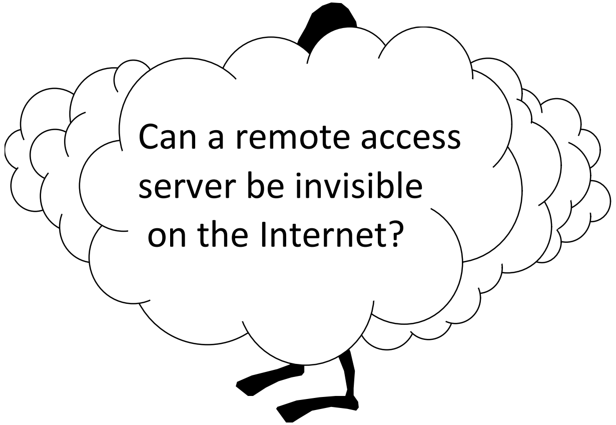


NAC – Network Access Control

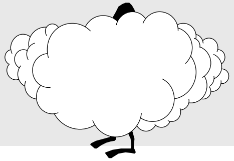
- NAC: Always check the user’s device before granting network access
  - Non-accepted devices can be connected to quarantine networks to update software
- Some products may support *identity-based access control* to networks
  - A client agent reports status to a network server
  - Example: Guest systems may be granted only Internet access from an access point, other systems full network access
  - Checks an be more granular and steer users access rights
- Policies can require secure remote VPN connections for certain services
  - Even used on internally on own networks!
- NAC violates Richard Stiennon's first (and only) rule of network security:
  - ”You shall not trust an end-point to report its own state”
  - System state can be spoofed, how would the server know?
- All unpatched systems meet on the quarantine network
  - A gold mine for an attacker
  - Servers on this network should not be able to access each other (see Link-level lecture)



Hiding systems and services



Hiding systems and services



- Can a service be offered to outside users, but still not be detected by unauthorized users?
  - Idea: An invisible system can not be a target for attacks (e.g SYN flooding)
- Robots constantly scan TLS, SSH and IPsec severs and guess username/passwords
  - Solution(?): Move service to a random port number (e.g. 47291)
  - A full port-scan will find it, but most scans will not
- Firewalls may allow only trusted IP addresses to connect (whitelisting)
  - Problem: All trusted IP addresses may not be known in advance
  - IP addresses can be faked but requires a MITM to create a TLS or SSH session
- Can we invent some kind of stealthy authorization mechanism?
  - Only responds to SYN from authorized users
  - Maybe we can give some additional authorization information with first SYN segment?
  - Or to let the server know in advance that we are coming?
  - Something which shows that we are authorized to talk to it?

Port knocking

- Way to instruct firewall that an authorized user is coming
  - Originally for Unix (2003) by Martin Krzywinski*
  - Today even iPhone and Android are supported
- User first sends SYNs to pre-defined port numbers
  - Correct sequence opens access to service from that IP address
  - No replies from server sent
- Example:
  - SYN→port 12324, SYN→port 58003, SYN→port 7419
  - Now **port 443 opens** for that IP address
  - Brute force (3 SYNs) requires 65,535^3 attempts = 10^{14} guesses
  - UDP can also be used for knocking
- Sequence can be different for different IP addresses
  - Sequence = hash (IP_addr, time, secret)
  - username can be sent as data in SYN segment
  - Time prevents replay attacks by a MITM
- Server needs software to dynamically configure its firewall
  - Listens to SYNs that are blocked by the firewall
  - When correct sequence detected, reconfigures filter for the service port
- Clients need software to do the knocking
- Benefits:
  - Only authorized users can see and connect to services
  - System is hidden from scans (stealthy) and can not be attacked
  - The service itself needs no modification
  - Bugs in protocols (IP stack) not possible to exploit
  - Offers some protection against zero-day vulnerabilities
- Malicious software can also use it to disguise its presence
  - Hides on network, wakes up after correct sequence
  - Own network scan will not detect the service

Krzywinski, M. 2003. Port Knocking: Network Authentication Across Closed Ports. SysAdmin Magazine 12: 12-17

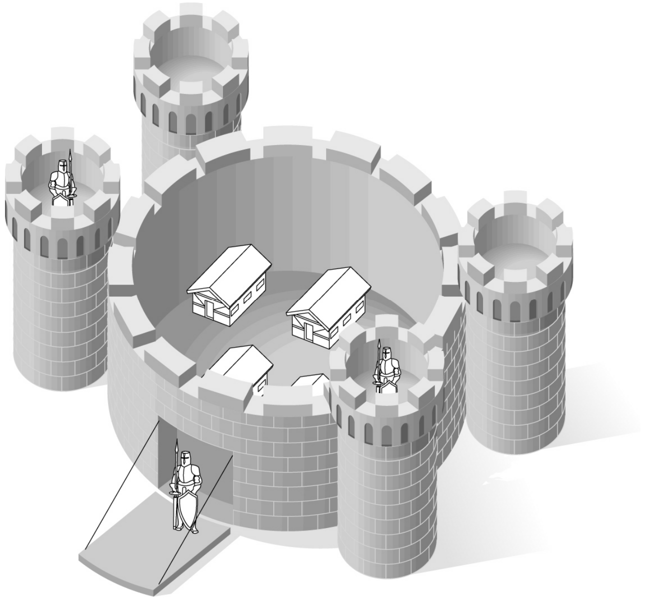
Single packet authorization (SPA)

- Developed 2005
- Same idea but only a single authorization packet is sent containing:

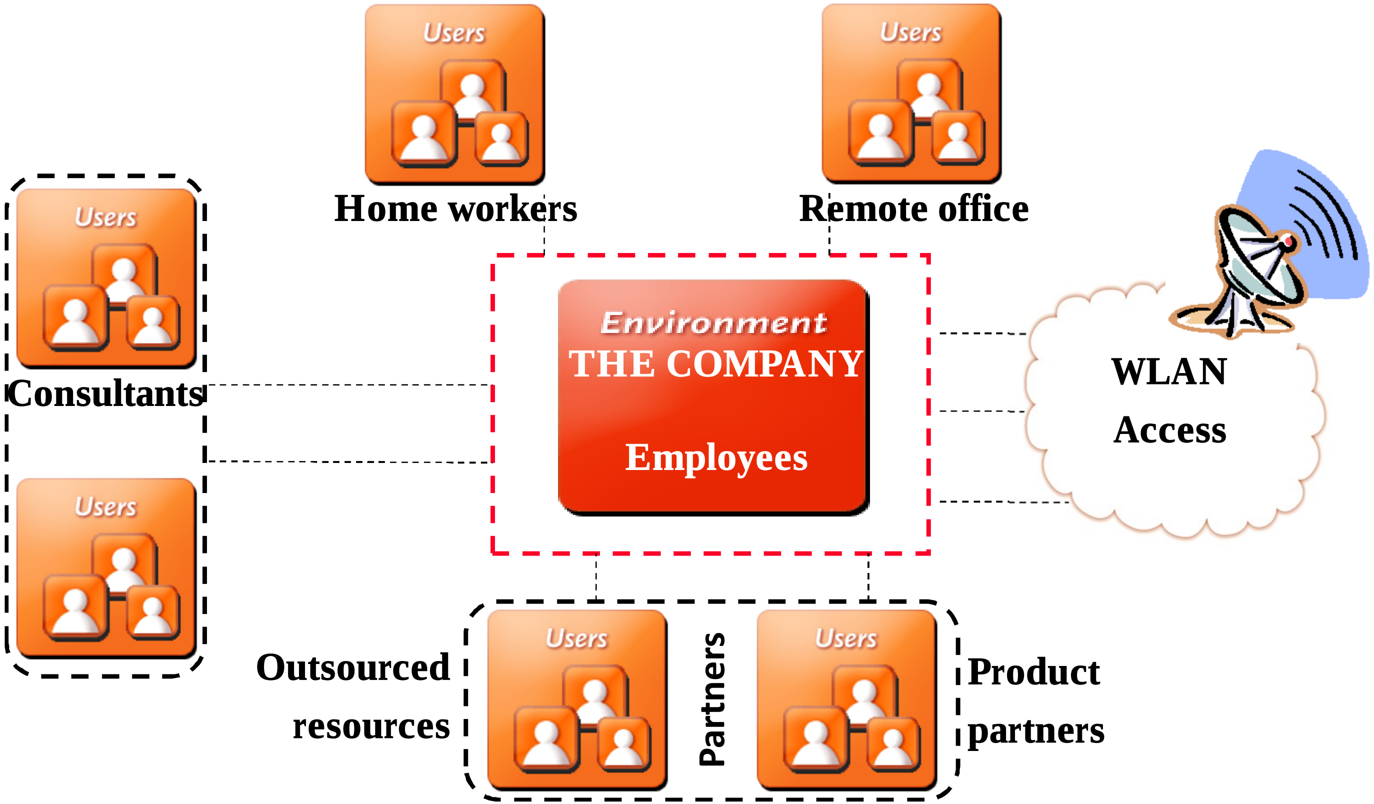
  username, hash (IP_address, username, time, secret)

- Public/private keys can also be used to verify users
- Practical issues
  - A new timestamp should always be required (for replay protection)
  - NAT may be a problem if IP address present in hash
  - Synchronized clocks needed (minute level ?)
  - For how long should the port be open? A web session may require many TCP connections
- Well-known implementation
  - FireWall KNock OPerator (fwknop) by Michael Rash (one UDP message)
  - https://www.cipherdyne.org/fwknop
  - Works and interacts with iptables on Unix/Linux systems

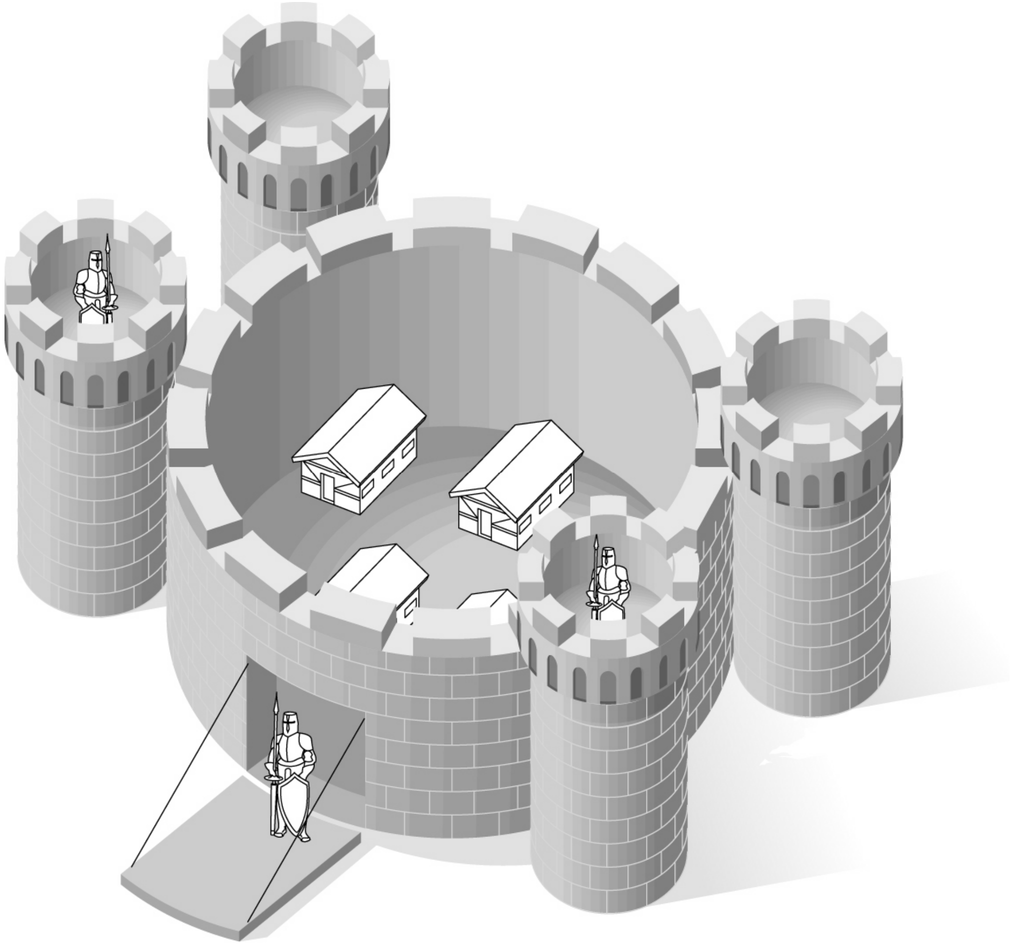
Network Design & Zero Trust Architectures



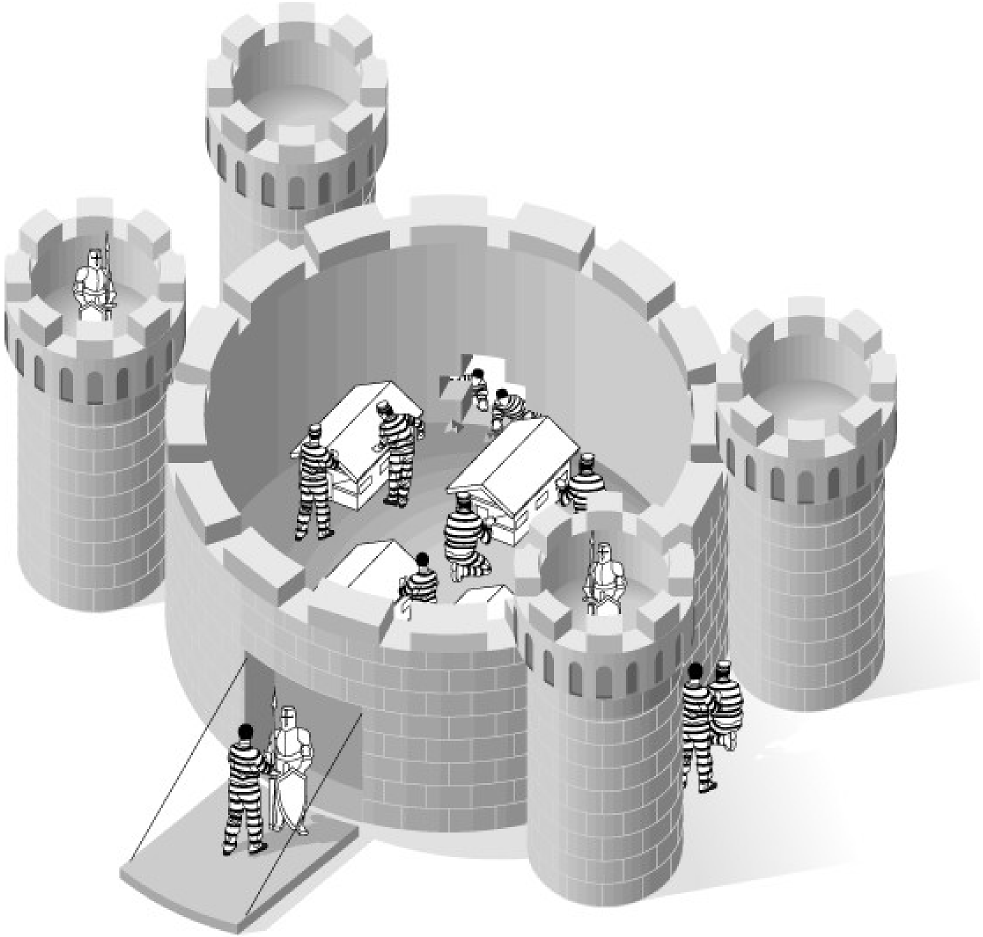
We can no longer hide behind a wall



A false sense of security



A gatekeeper is not enough!



Next Generation Firewalls

Making Everything Easier!™

Palo Alto Networks 2nd Edition

Next-Generation Firewalls for DUMMIES

A Wiley Brand

Learn to:
- Differentiate between "good" and "bad" applications
- Identify evasive techniques used by applications and modern threats
- Implement effective application and network controls

Brought to you by paloalto

Lawrence C. Miller, CISSP

Free e-book

Over time, a traditional central firewall becomes full of holes

We need an application and user centric approach:

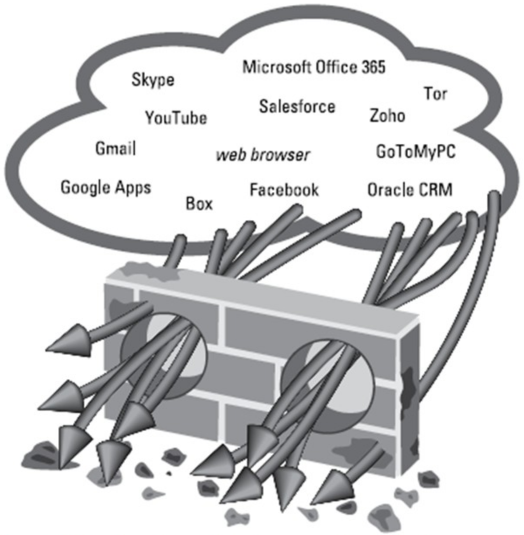


Figure 3-1: Port-based firewalls can't see or control applications.

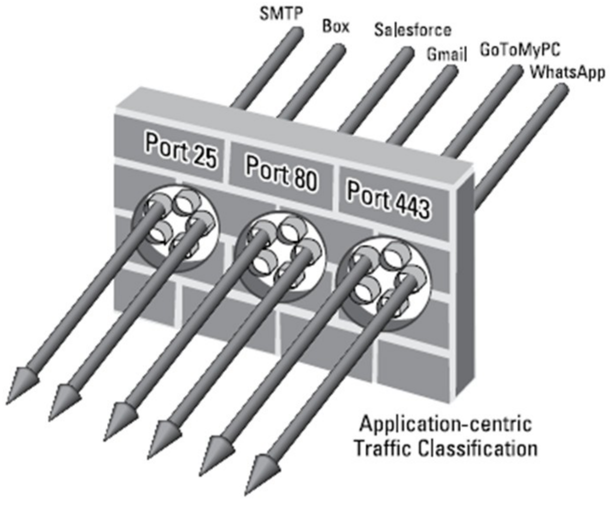


Figure 4-1: Application-centric traffic classification identifies specific applications flowing across the network, irrespective of the port and protocol in use.

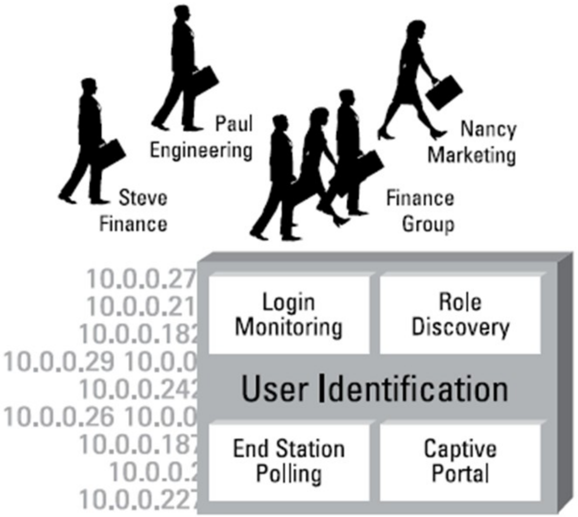
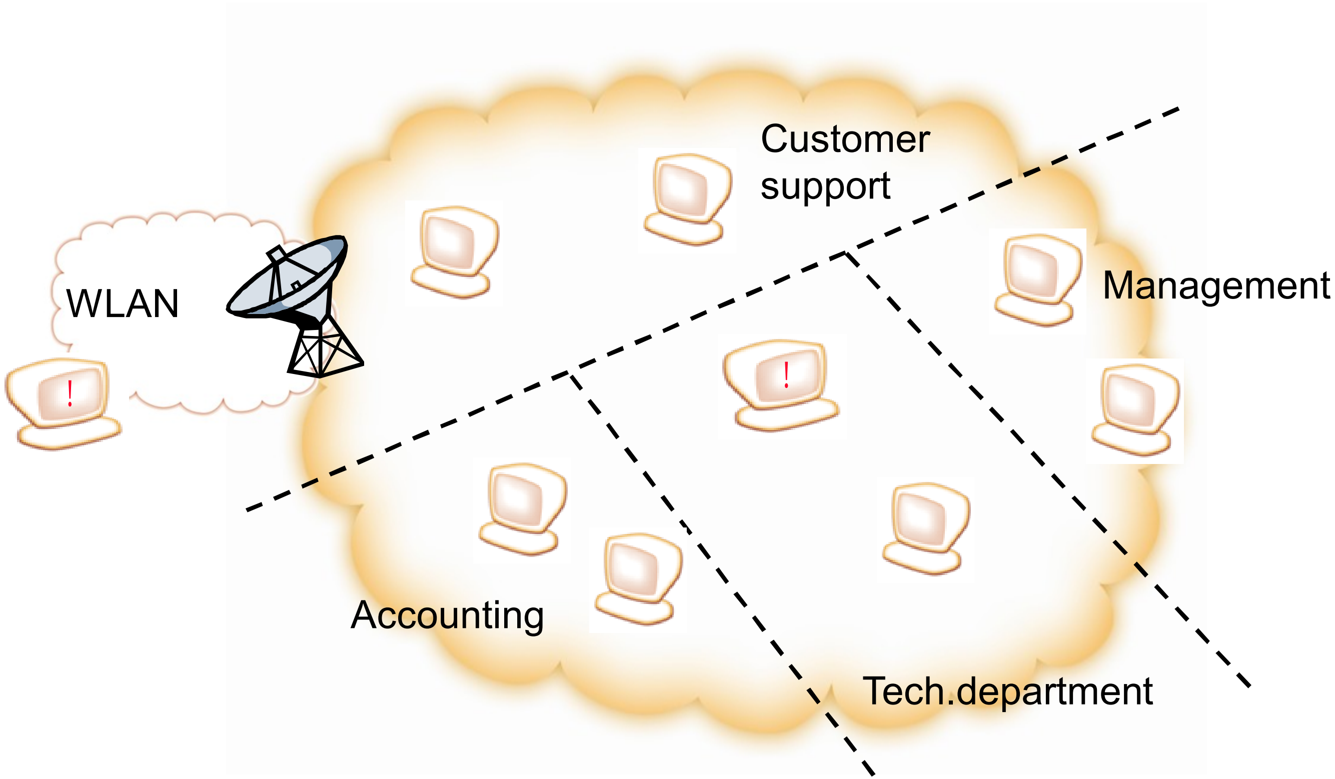


Figure 4-3: User identification integrates enterprise directories for user-based policies, reporting, and forensics.

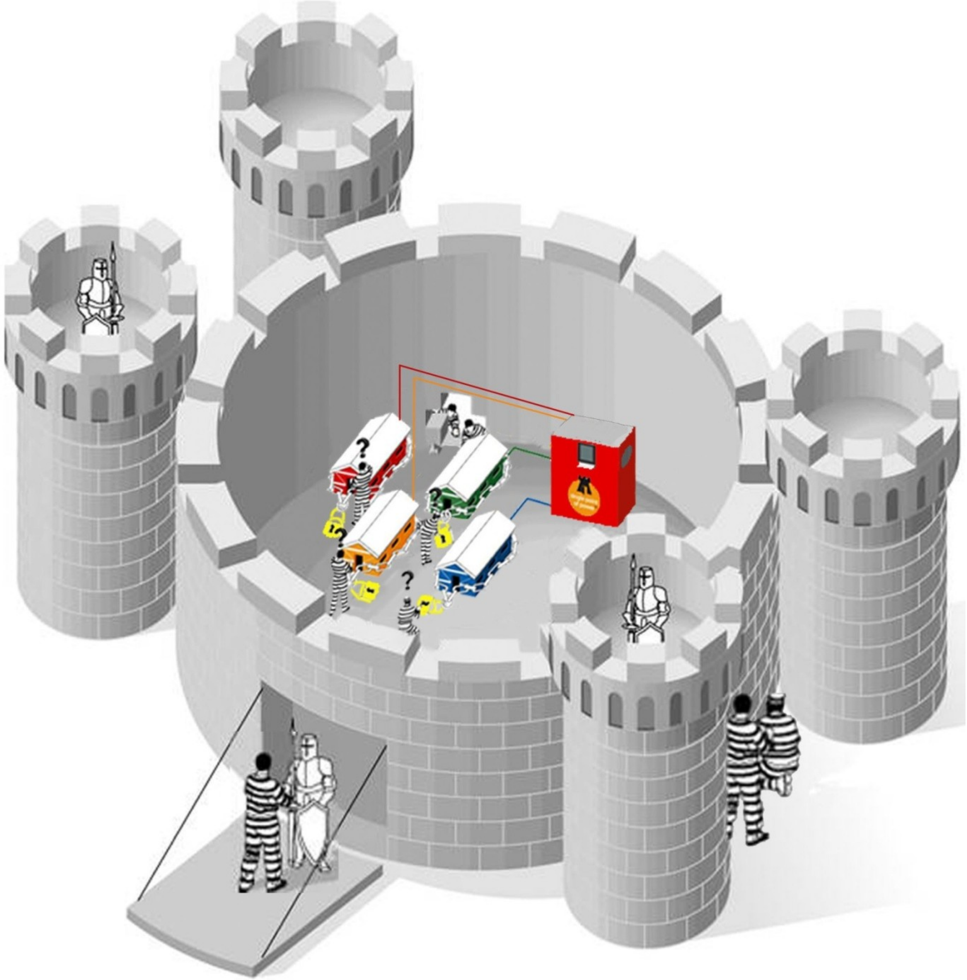
Internal segmentation is a first step



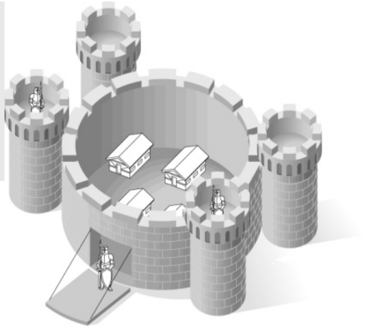
Next step: Protect the assets, not the gate

Protection at the source means that it doesn't matter how you reach the inside!

Solves both local and remote access!

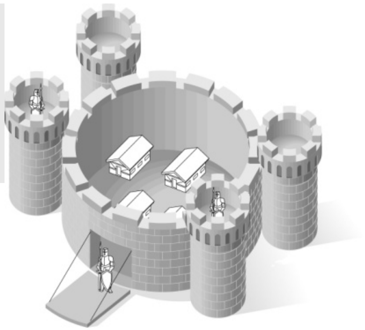


The Jericho Forum

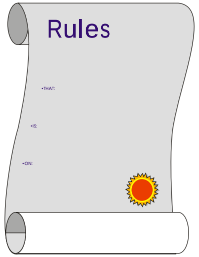


- An international forum of IT customer and vendor organizations
  - https://www.opengroup.org/forum/security
  - Royal Mail, Standard Chartered Bank, the BBC, …
- Mission: De-perimeterization – 2004
  - “Perimeter security has become obsolete”
  - “Today's corporate perimeters are full of holes, so information security solutions are needed that are effective in this increasingly de-perimeterized world.”
  - “The old hard-shell model of security isn't sustainable in light of the need for businesses to open up their networks to partners, consultants and clients”
- This way to think then resulted in the concept “Zero Trust”

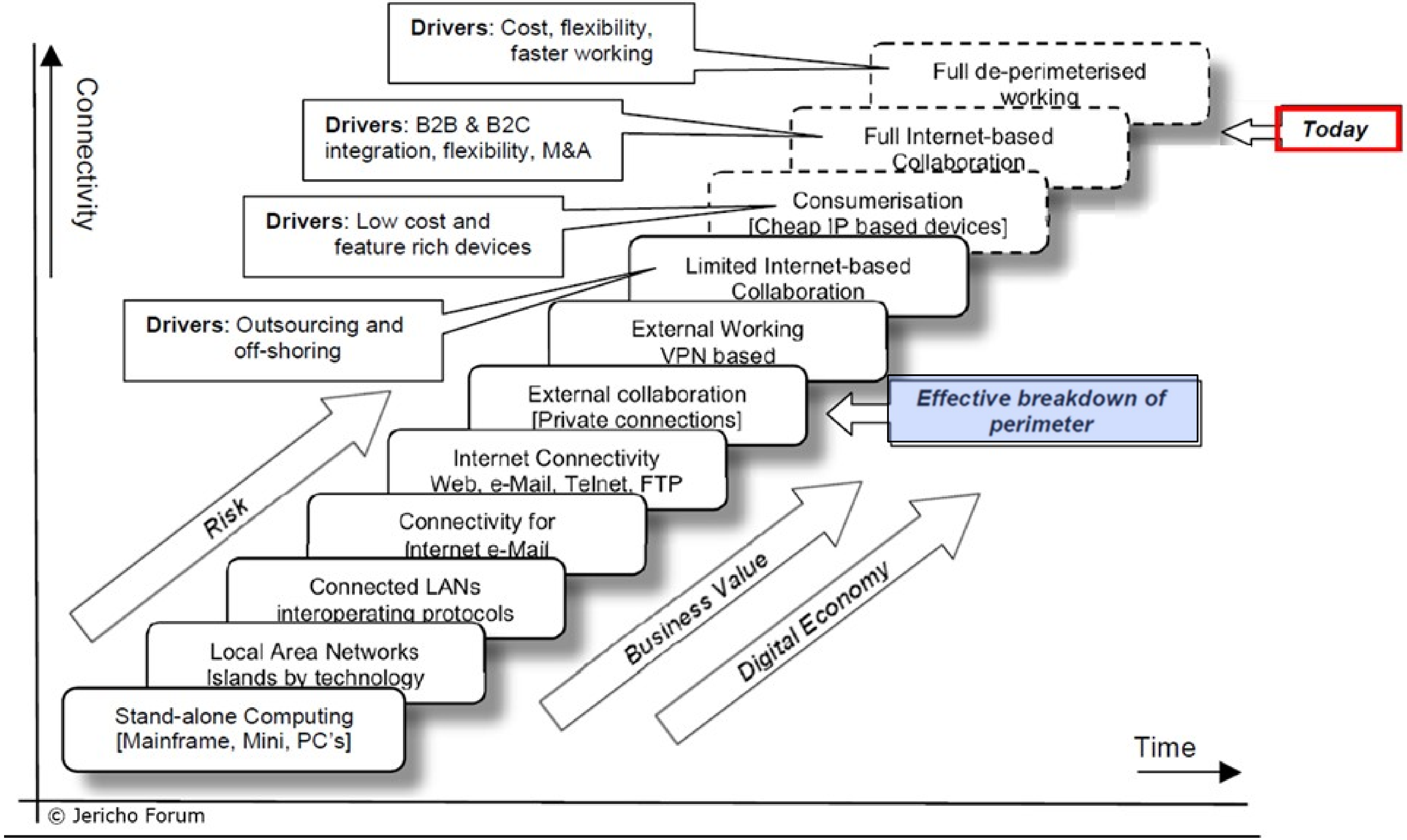
What is de-perimeterization?



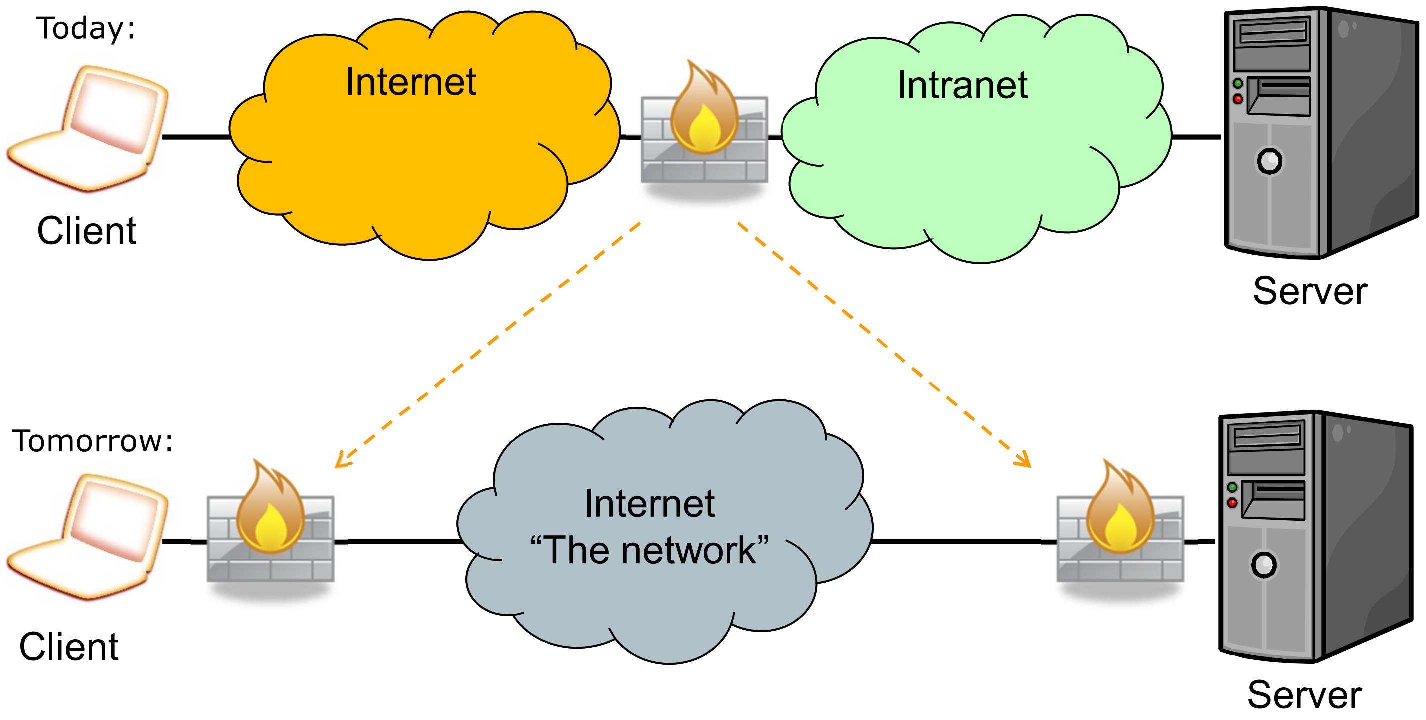
- Move security control closer to the end points
- Make access ”seamless” and base it on cooperation between applications and users
- Be in total control of all users’ access rights
- Add policies that dictate how and under what circumstances *<u>each user</u>* can access *<u>each service</u>*
- De-perimeterization doesn't necessarily mean discarding the firewall



De-perimeterization: re-think about firewalls

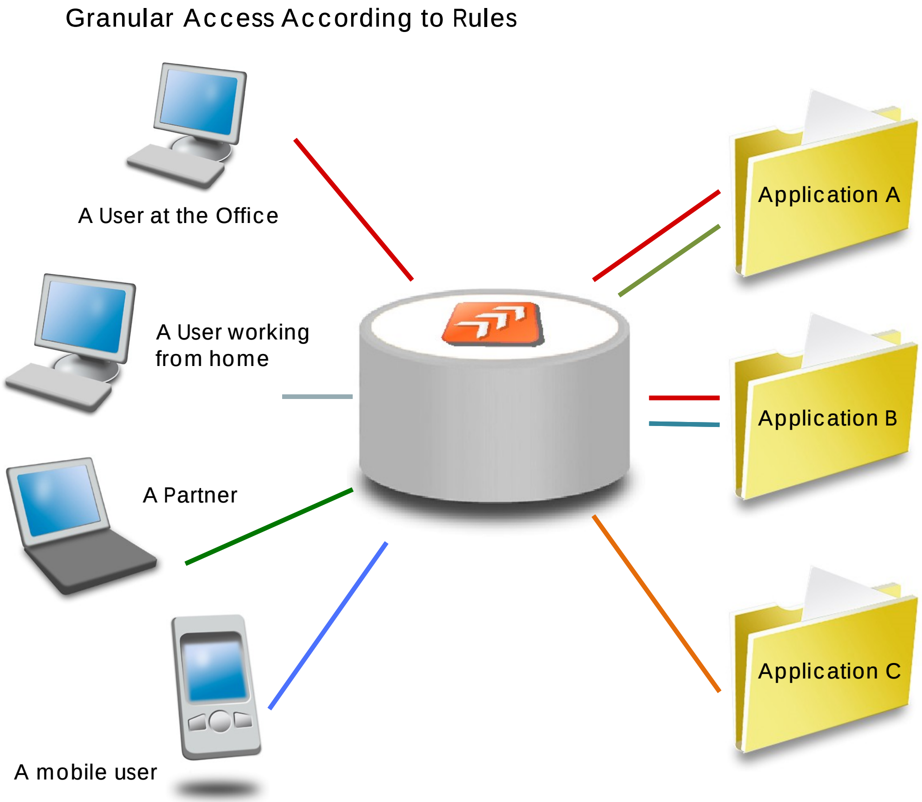

Drivers: Cost, flexibility, faster working
Full de-perimeterised working
Drivers: B2B & B2C integration, flexibility, M&A
Full Internet-based Collaboration
Today
Drivers: Low cost and feature rich devices
Consumerisation [Cheap IP based devices]
Limited Internet-based Collaboration
Drivers: Outsourcing and off-shoring
External Working VPN based
External collaboration [Private connections]
Effective breakdown of perimeter
Internet Connectivity Web, e-Mail, Telnet, FTP
Connectivity for Internet e-Mail
Connected LANs interoperating protocols
Local Area Networks Islands by technology
Stand-alone Computing [Mainframe, Mini, PC's]
Connectivity
Risk
Business Value
Digital Economy
Time
© Jericho Forum

Move protection to the end-points



✓ Remote access should not be different than other types of access

Role-based access control



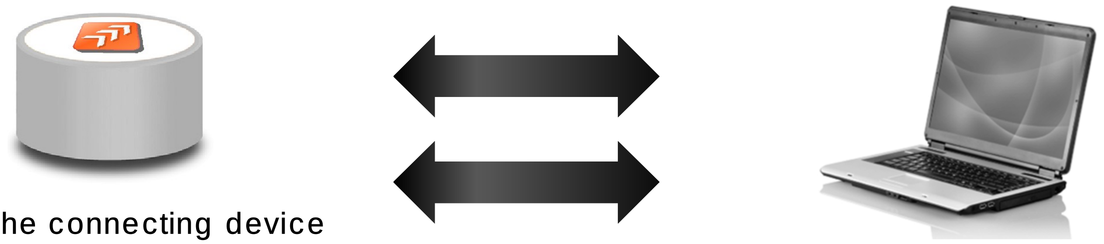
Full Access to the network for a trusted user on a trusted workstation

Very limited access from a mobile phone

Maybe just a file for a customer?

Just one page on the intranet for a partner?

End-point security: check the devices



Check the connecting device

Extensive checks should be supported: operating system, version, anti-virus, installed software, system configuration, sharing, connected devices...

Access should depend on outcome of security checks

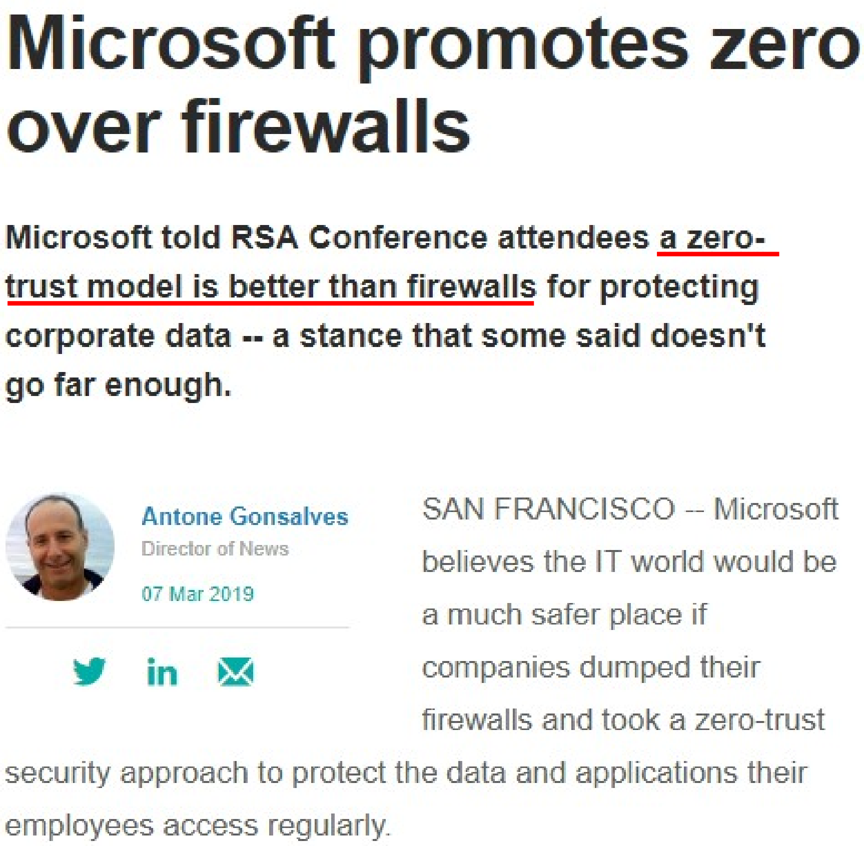
It is not a question about full access or no access!

Zero Trust - NIST

National Institute of Standards and Technology
U.S. Department of Commerce

- Zero trust (ZT) move defenses from static, network-based perimeters to focus on users, assets, and resources
  - Zero trust is de-perimiterisation taken one step further
- Zero trust focuses on protecting resources and not network segments, as the network location is no longer seen as the prime component
- Zero Trust assumes there is no implicit trust based solely on physical or network location (inside or outside) or based on asset ownership
- Principles of zero trust:
  - All communication is secured regardless of network location
  - Access to individual enterprise resources is granted on a per-session basis
  - Access to resources is determined by dynamic policy—including the observable state of client identity, application/service, and the requesting assets

https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf



Microsoft promotes zero-trust security over firewalls

Microsoft told RSA Conference attendees a zero-trust model is better than firewalls for protecting corporate data -- a stance that some said doesn't go far enough.

Antone Gonsalves
Director of News
07 Mar 2019

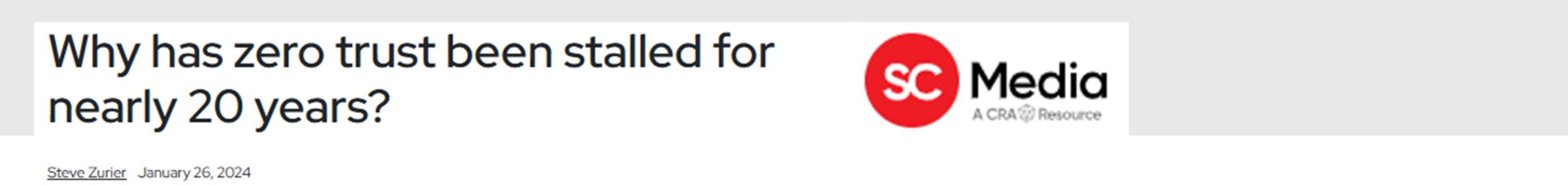
SAN FRANCISCO -- Microsoft believes the IT world would be a much safer place if companies dumped their firewalls and took a zero-trust security approach to protect the data and applications their employees access regularly.

On Wednesday [March6, 2019] Microsoft told RSA Conference attendees that firewalls are no longer useful as a first line of defense.

What has made the trusted technology obsolete is the variety of devices employees use to access corporate data from far-flung places outside corporate offices.

Employees also no longer seek entry to applications sitting only in private data centers. Today, business software could just as easily live in a public cloud or exist as an online service.

https://searchsecurity.techtarget.com/news/252459059/Microsoft-promotes-zero-trust-security-over-firewalls



Why has zero trust been stalled for nearly 20 years?

Steve Zurier January 26, 2024

- More than two decades have passed and still less than 1/3 of organizations that have implemented it

- Reasons:
  - High costs, hard to show ROI
  - Complexity to integrate into workflows

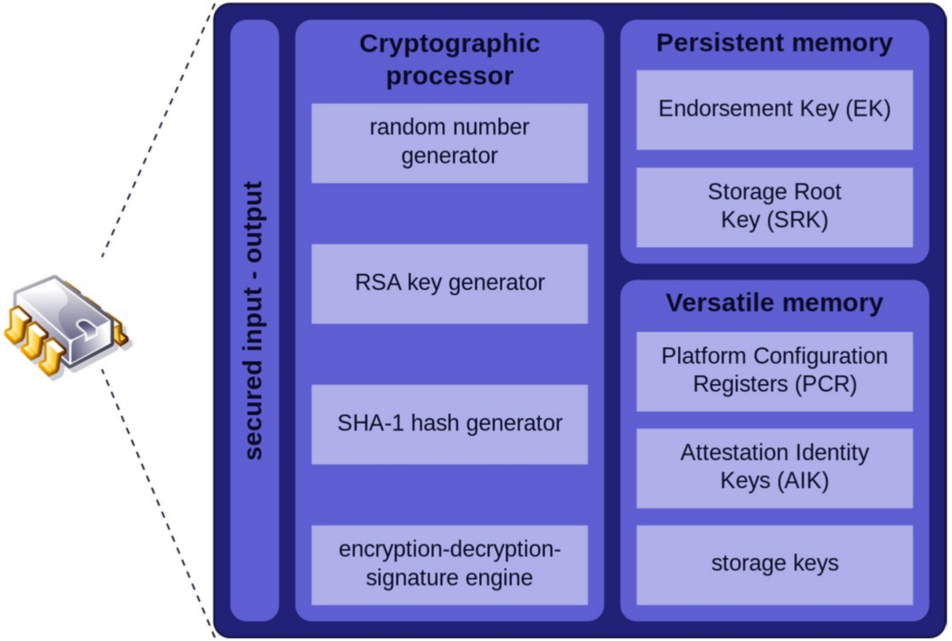
- 62% believe that it in becoming increasingly important

https://www.scmagazine.com/research-article/why-has-zero-trust-been-stalled-for-nearly-20-years

Device Attestation (DA)

- How can a device prove its identity and that it is not compromised, i.e. executes the intended code?
  - Can a device evaluate itself?
- Possible with hardware support
  - TEE: Trusted Execution Environments
  - Different assurance levels depending on hardware
- Platforms (with different functionality):
  - INTEL Software Guard Extensions (SGX)
  - AMD Secure Encrypted Virtualization (SEV)
  - ARM Trust Zone
  - RISC-V (open source to show what is really needed…)

TEE – Trusted Execution Environments



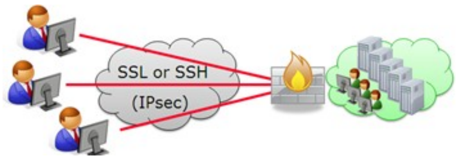
Trusted Platform Module, TPM (picture from Wikipedia)
External module, not part of the processor in the system

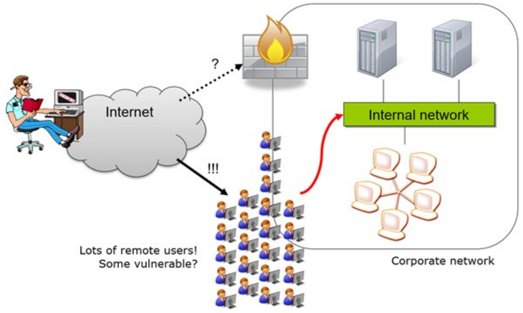
EK = RSA/ECC certificate from manufacturer of the system

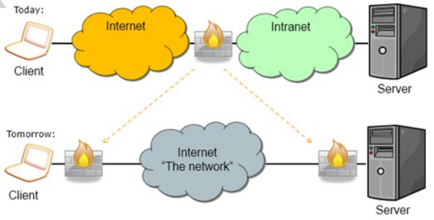
- Processor or computer contains a TEE
- Performs a secure boot and is isolated from the rest of the system
- Device attestation can now be done by a fully trusted application
- TEE and TPM devices can perform:
  - data signing
  - signature verification
  - checks of main processor, memory and connected hardware
  - support secure boot

Summary

- Virtual private networks – VPN
  - Site to site, user to site
  - Application-level (TLS, SSH) vs. network level (IPsec)
  - Full remote access from public and home networks dangerous
- Port knocking – hiding servers
- Zero trust and the Jericho Forum
  - Zero trust = move protection to end-points and focus on users, assets and resources
  - NAC – dynamic network access control based on user role, authentication system, computer configuration, location, etc.
- Trusted execution environments can be used to verify system integrity


SSL or SSH
(IPsec)


Internet
?
!!!
Internal network
Lots of remote users!
Some vulnerable?
Corporate network


Today:
Internet
Intranet
Client
Server
Tomorrow:
Internet
"The network"
Client
Server