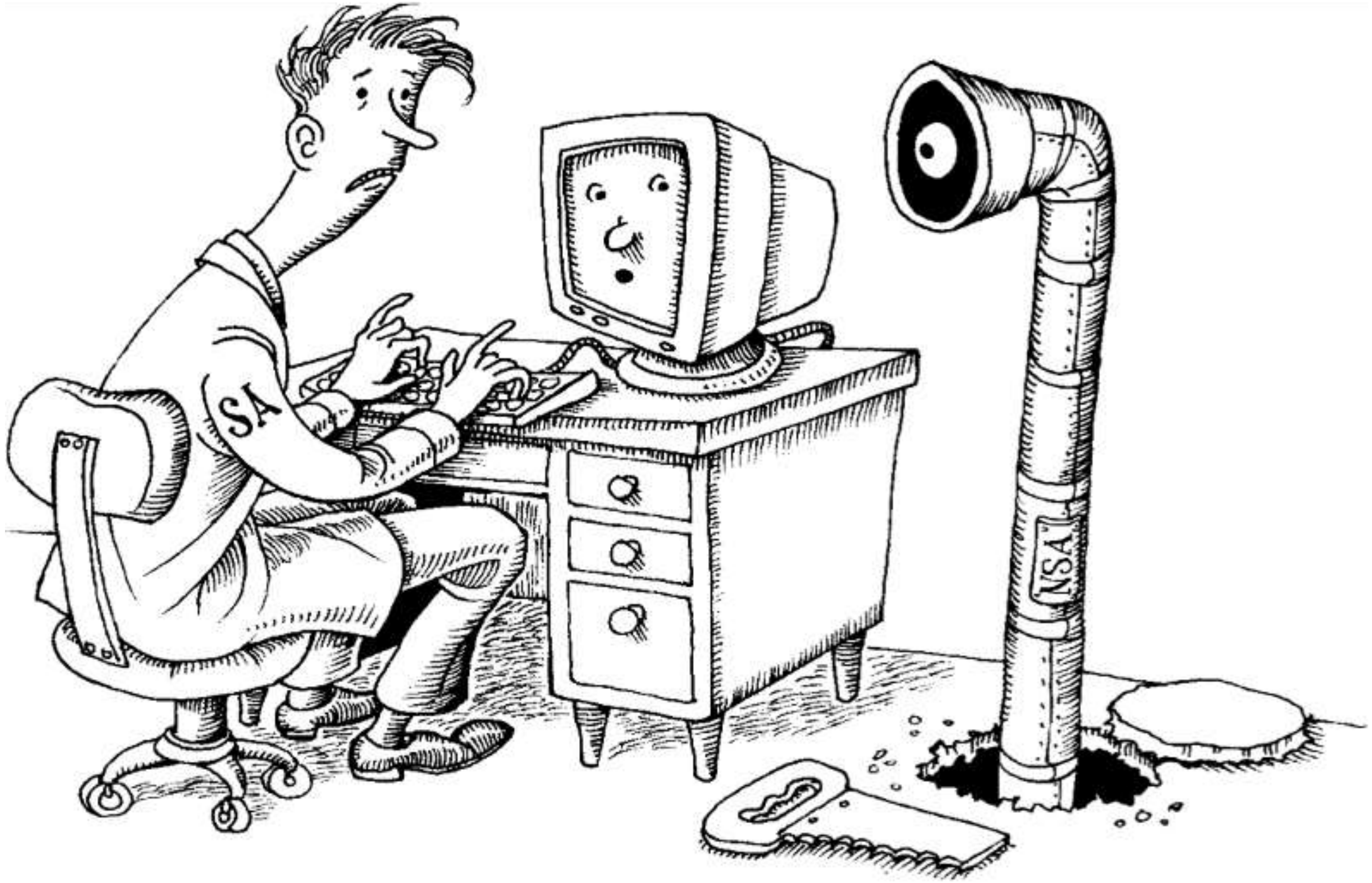
CS183

Instructor: Ali Davanian



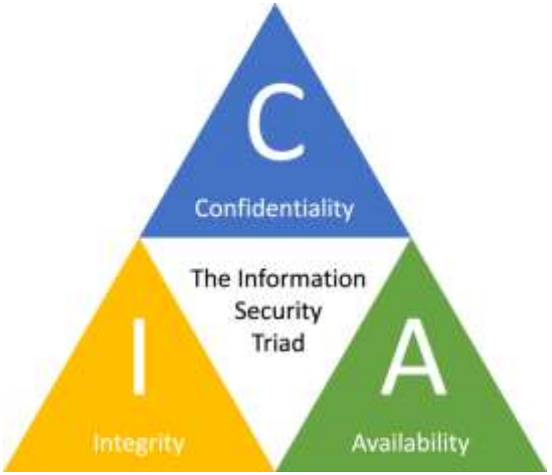
Security

Logistics

- This is our last lecture
  - We studied more than 11 chapters of the textbook
- Next week is your presentations
  - Sign up (Link on Slack and iLearn)
  - Make it interesting for your classmates
- Please evaluate me
  - Please be polite and fair in your evaluations
  - Instructors are not able to access results until grades have been submitted
  - https://ieval.ucr.edu/

Chapters	Topic
Chapter 2	Booting and System
Chapter 7	**Scripting and the Shell**
Chapter 3	Access Control
Chapter 4	Process Control
Chapter 5	The Filesystem
Chapter 6	Software Installation/ Management
Chapter 8	User Management
Chapter 13	TCP/IP Networking
Chapter 15	IP Routing
Chapter 16	DNS: Domain Name System
Chapter 27	Security
Partially touched: 1 and 14	Introduction and Physical Networking

Elements of security - ”CIA Triad”



- CIA or Triangle of Security stands for:
  - Confidentiality: it is about privacy of data
  - Integrity: it relates to the authenticity of information
  - Availability: it expresses the idea that information must be accessible to authorized users when they need it
- Security is a process!
  - Consider the CIA principles as you design, implement, and maintain systems and networks

How security is compromised?

- **Social engineering**: in the context of information security, it is the psychological manipulation of people into performing actions or divulging confidential information (Wikipedia).
- **Software vulnerabilities**: they are security-sapping bugs like Buffer overflows.
- **Denial-of-service (DoS) attacks**: A DoS attack aims to interrupt a service or adversely impact its performance, making the service unavailable to users.
- **Insider attacks**: when Employees, contractors or consultants abuse their privileges to reveal data, disrupt systems for financial gain, or create havoc for political reasons
- **Misconfiguration**: a not-so-secure configuration that allows hackers to break in.

Basic Security Measures

- **Software Updates**: keeping systems updated with the latest patches
- **Backups:** keeping a copy of system states and data in case there is a breach
- **Removing Unnecessary services**: disable (and possibly remove) those services are unnecessary, especially if they are network daemons.
- **Password measures**: enforce password aging, complexity, time out etc.
- **Penetration testing:** lay the hacker role and try to break in your network; if you found a backdoor, close it
- **Firewalls:** filter out unnecessary traffic to your network
- **Anti Virus:** anti Viruses are usually the last line of defense; if the malware bypassed firewalls (and other measures), the antivirus can still detect an clean it
- **Logging:** log and store events as they will be very useful to detect, and analyze attacks

Security tools

- **Pen**etration testing
  - **Nmap**: network port scanner
  - **Metasploit**: penetration testing software
  - **Nessus**: scans network for services, and can check their vulnerability
- Security auditing and compliance checking:
  - **Lynis:** It performs an extensive health scan of your systems to support system hardening and compliance testing
- Password Complexity:
  - **John the Ripper:** implements various password-cracking algorithms in a single tool. Direct John to the file to be cracked e.g./etc/shadow
  - **THC Hydra:** similar but works online

Network Intrusion Detection Systems (NIDS)

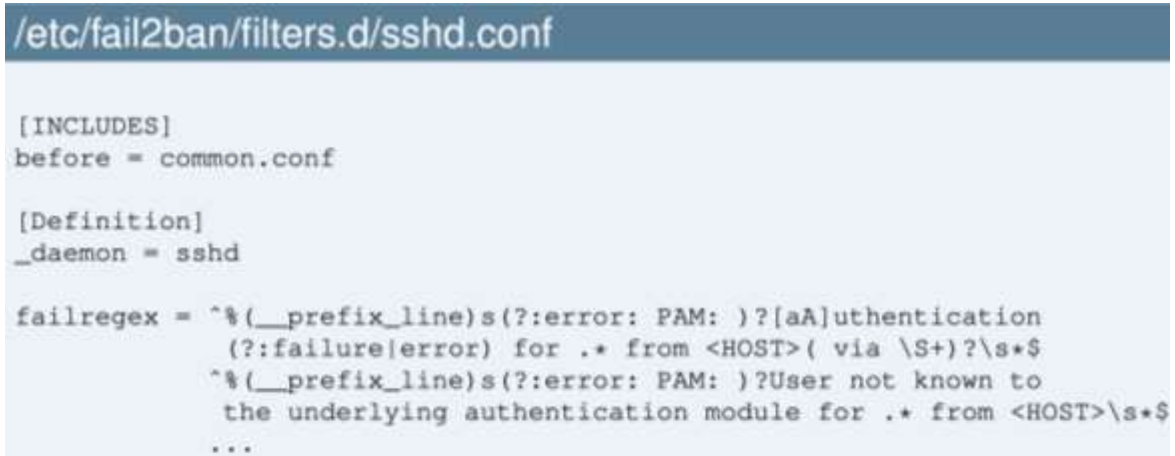
- **Snort:** the popular network intrusion detection/prevention system
  - Snort IPS uses a series of rules that help define malicious network activity and uses those rules to find packets that match against them and generates alerts for users
  - alert tcp !192.168.1.0/24 any -> 192.168.1.0/24 111 (content: "|00 01 86 a5|"; msg: "external mountd access";)
- **Bro (Zeek):** the programmable network intrusion detection system
  - Zeek is a powerful network analysis framework; compared to SNORT Zeek/Bro is more of a passive approach.
- **Suricata**: very similar to SNORT but offers a scripting language (LuaJIT), and was designed with multi-threading (and performance) in mind

Host Based Intrusion Detection

- A host-based IDS is an intrusion detection system that monitors the computer infrastructure on which it is installed
- OSSEC is a scalable, multi-platform, open source Host-based Intrusion Detection System (HIDS). Some features are:
  - Log file analysis
    - Collect logs from files (for instance “/var/log/messages”)
    - Alerting when output of a command changes (for instance “netstat -tan |grep LISTEN”)
  - Filesystem integrity checks
    - It does that by looking for changes in the MD5/SHA1 checksums of the key files in the system
  - Root kit detection
    - Look for hidden processes, access to important files owned by root etc.

Fail2Ban: brute-force attack response system

- Fail2Ban scans log files like /var/log/auth.log and bans IP addresses conducting too many failed login attempts
- Fail2Ban comes out-of-the-box ready to read many standard log files, such as those for sshd and Apache
  - It can read any file of your choice

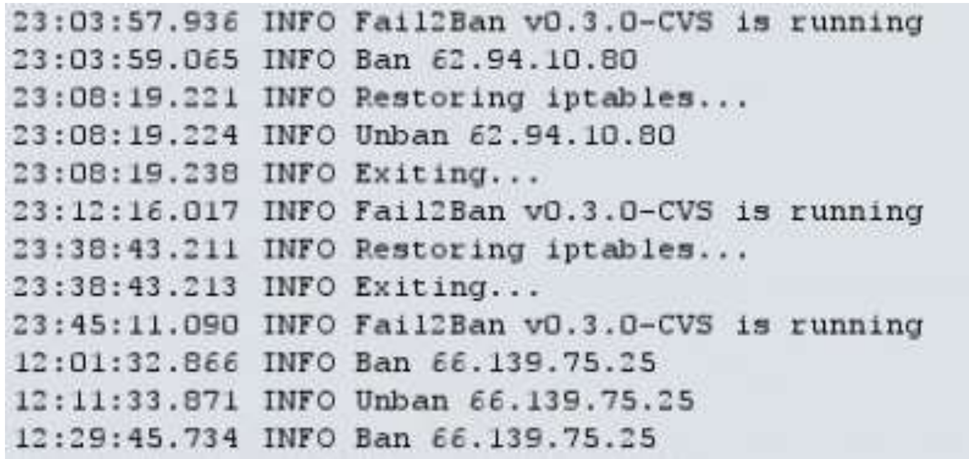


/etc/fail2ban/filters.d/sshd.conf

```
[INCLUDES]
before = common.conf

[Definition]
_daemon = sshd

failregex = ^%(__prefix_line)s(?:error: PAM: )?[aA]uthentication
             (?:failure|error) for .* from <HOST>( via \S+)?\s*$
            ^%(__prefix_line)s(?:error: PAM: )?User not known to
             the underlying authentication module for .* from <HOST>\s*$
            ...
```



```
23:03:57.936 INFO Fail2Ban v0.3.0-CVS is running
23:03:59.065 INFO Ban 62.94.10.80
23:08:19.221 INFO Restoring iptables...
23:08:19.224 INFO Unban 62.94.10.80
23:08:19.238 INFO Exiting...
23:12:16.017 INFO Fail2Ban v0.3.0-CVS is running
23:38:43.211 INFO Restoring iptables...
23:38:43.213 INFO Exiting...
23:45:11.090 INFO Fail2Ban v0.3.0-CVS is running
12:01:32.866 INFO Ban 66.139.75.25
12:11:33.871 INFO Unban 66.139.75.25
12:29:45.734 INFO Ban 66.139.75.25
```

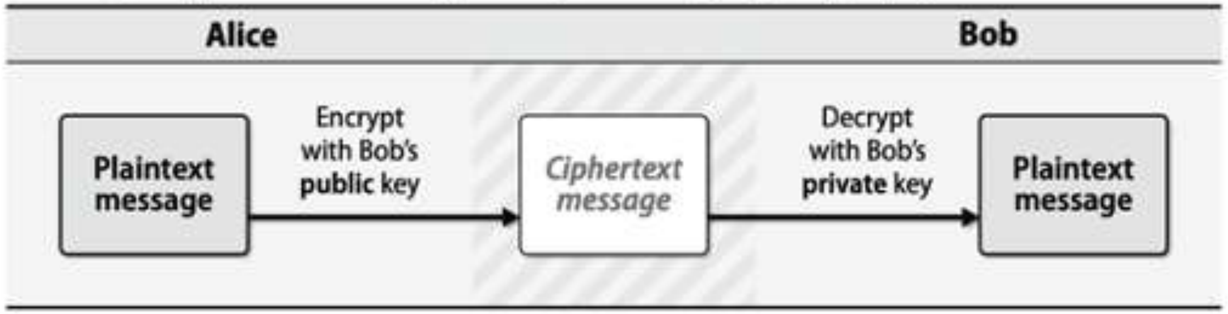
Cryptography

- In the cryptography context, “Alice” and “Bob” are at the two sides of the communication and they want to securely communicate with:
  - *Confidentiality:* messages are impossible to read for everyone except the intended recipients.
  - *Integrity*: it is impossible to modify the contents without detection.
  - *Non-repudiation:* the authenticity of the message can be validated.
- Symmetric cryptography or “classic” cryptography
  - Alice and Bob share a secret key that they use to encrypt and decrypt messages.
  - **Limitation:** Alice and Bob need to securely exchange the secret key in advance

Public key cryptography

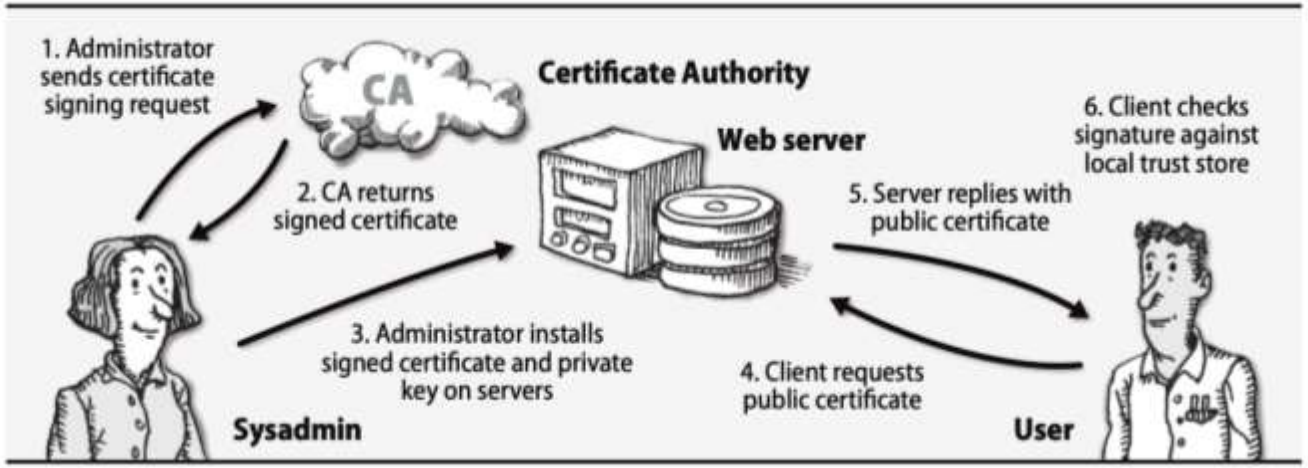
- The idea is keeping the keys for decryption private while sharing the key for encryption readily available
  - A message sent for “Bob” can only be read by “Bob”
  - Useful for cases where parties want to communicate for the first time
- How should Alice know the public key belongs to “Bob”? What if “Mallory” claims to be Bob and present a fake public key?
  - There needs to be a signature validation process

Exhibit A **Sending a ciphertext message with public key cryptography**



Public key infrastructure process for the web

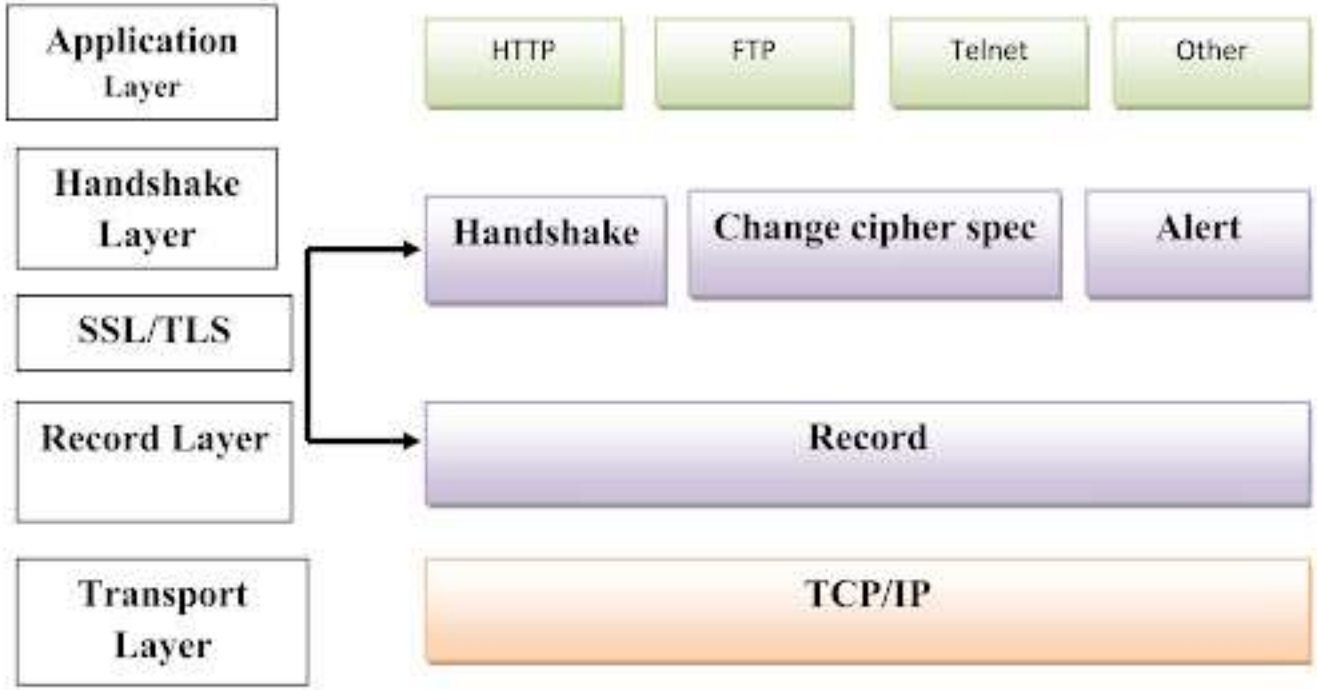
Exhibit B **Public key infrastructure process for the web**


1. Administrator sends certificate signing request
CA
Certificate Authority
Web server
2. CA returns signed certificate
6. Client checks signature against local trust store
5. Server replies with public certificate
3. Administrator installs signed certificate and private key on servers
4. Client requests public certificate
Sysadmin
User

- The “Certificate Authority” (CA) is implicitly trusted in this system
  - Examples are such as GeoTrust and VeriSign
- The user can check the authenticity of signatures because operating systems are shipped with CA certificates
  - Signing is another interesting cryptography context

Transport Layer Security (TLS)

- Transport Layer Security (TLS) uses public key cryptography and PKI to secure messages between nodes on a network



openssl

- openssl is an administrator's TLS multitool:
  - generate public/private key pairs
    - openssl genrsa -out admin.com.key 2048
  - create a certificate signing request
    - openssl req -new -sha256 -key admin.com.key -out admin.com.csr
  - examine the cryptographic properties
    - openssl x509 -noout -text -in google.com.pem
    - openssl s_client -connect google.com:44
  - encrypt and decrypt files
  - create certificate authorities
  - myriad other cryptographic operations
- It is open source and it has been examined by thousands of people

The Secure Shell - SSH

- It is a protocol for remote logins and for securing network services on an insecure network
  - Remote command execution
  - Shell access
  - File transfer
  - Port forwarding
  - Network proxy services
  - Even VPN tunneling
- SSH is a client/server protocol that uses cryptography for authentication, confidentiality, and integrity of communications

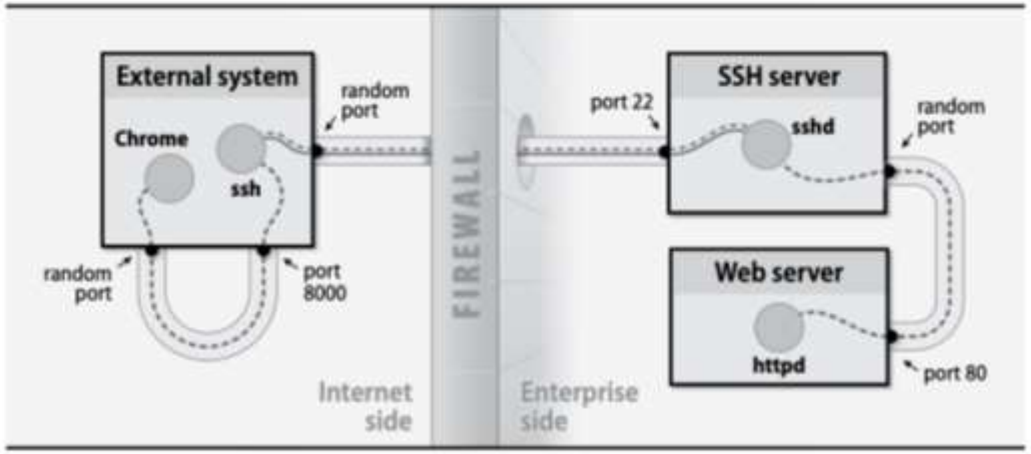
OpenSSH - Open source SSH implementation

- **ssh**, the client
- **sshd**, the server daemon
- **ssh-keygen**, for generating public/private key pairs
- **ssh-add** and **ssh-agent**, tools for managing authentication keys
- **ssh-keyscan**, for retrieving public keys from servers
- **sftp-server,** the server process for file transfer over SFTP
- **sftp** and **scp**, file transfer client utilities

ssh port forwarding

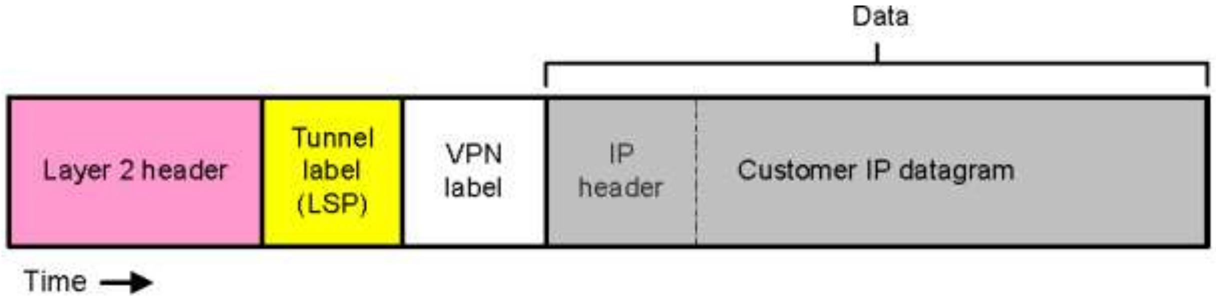
- Alice wants to establish an HTTP connection to a web server on an enterprise network an access to port 80 is blocked by the firewall
- Alice can route the connection through the SSH server
  - Alice can use SSH port forwarding feature to access the web server
  - ssh -L 8000:webserver:80 server.admin.com
  - Alice can now open the webpage in her browser by specifying port 8000

Exhibit D An SSH tunnel for HTTP

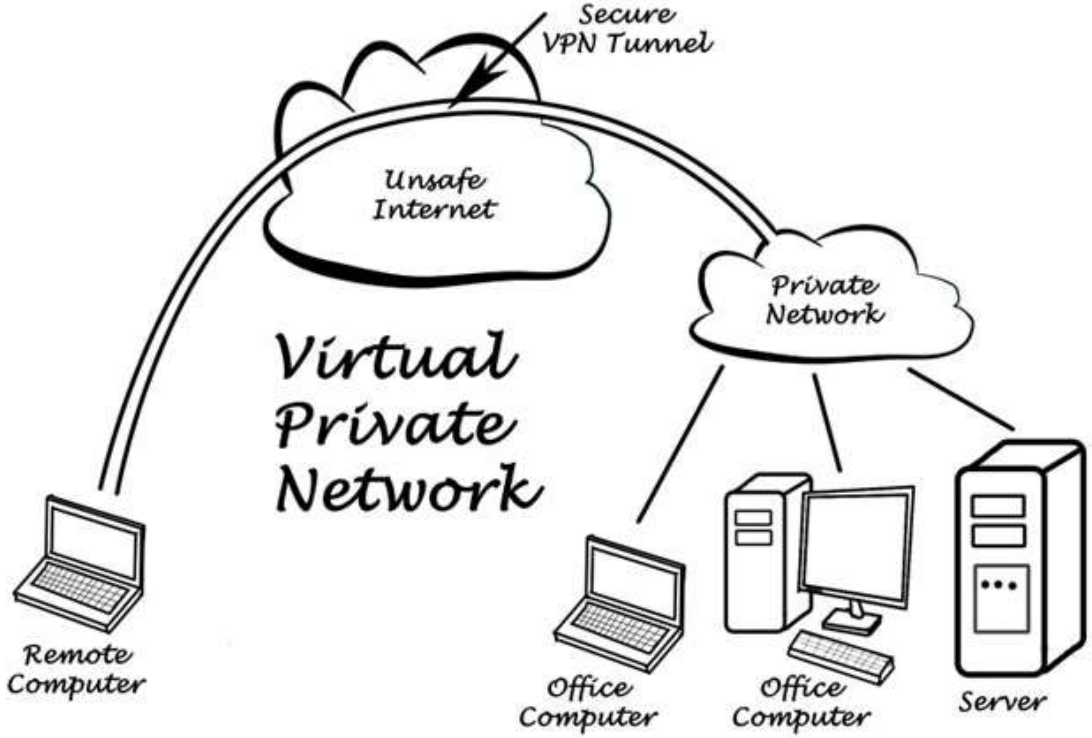


Virtual private networks (VPNs)

- A VPN is a connection that makes a remote network appear as if it were directly connected
- Internet Protocol security (IPsec) is an approved, authentication and encryption system
- Linux and FreeBSD include native kernel support for IPsec



A. Layer 3 data encapsulation in accordance to RFC 2547bis



Questions?

Further reading

- Computer Security Textbook (taught in UC Berkeley)
- Google Security Engineering Interview Study notes
- Mitre Attack Matrix for Enterprise