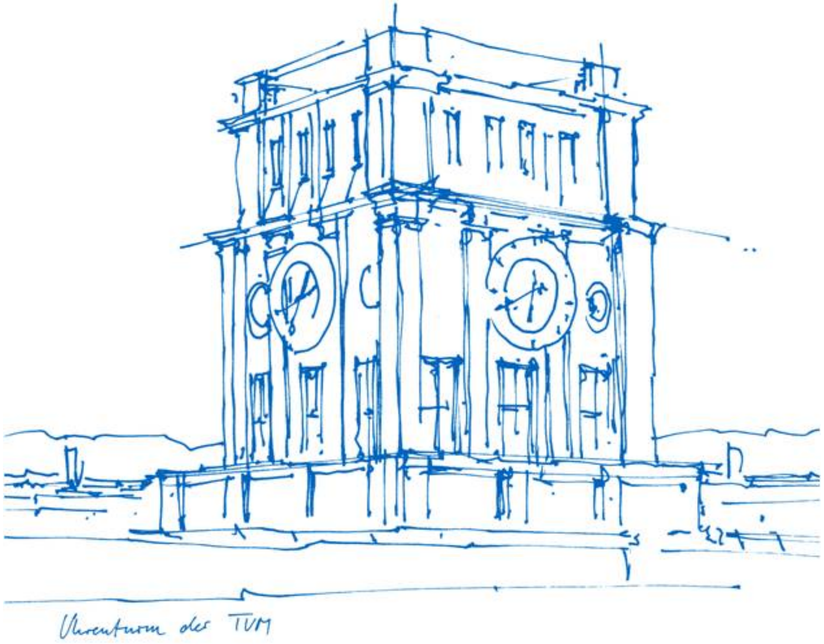
Network Security WS17/18

Attacks and Attack Detection

Quirin Scheitle



Attacks and Attack Detection

Have you ever been attacked (in the IT security sense)?

What kind of attacks do you know?

What can happen?

❑ Part 0:	Attacks
❑ Part I:	Attack Prevention
❑ Part II:	Attack Detection
❑ Part III:	Response Mechanisms
❑ Part IV:	Recent Attacks and Research

Attacks Types

Disruptive:

The goal is to fully deny the victim‘s service to its clients

Degrading:

Part of the victim’s resources (e.g. 30%) are occupied by attackers.

Can remain undetected for a signification time period

Customers experience slow response times or now service during high load periods. → Customers go to an other Service Provider.

Data Exfiltration:

Confidential data, passwords, password files, keys, …

Control:

Being able to command a machine (may not interfere with normal operation), possibly “lie low” for extended periods of time

System Vulnerabilities

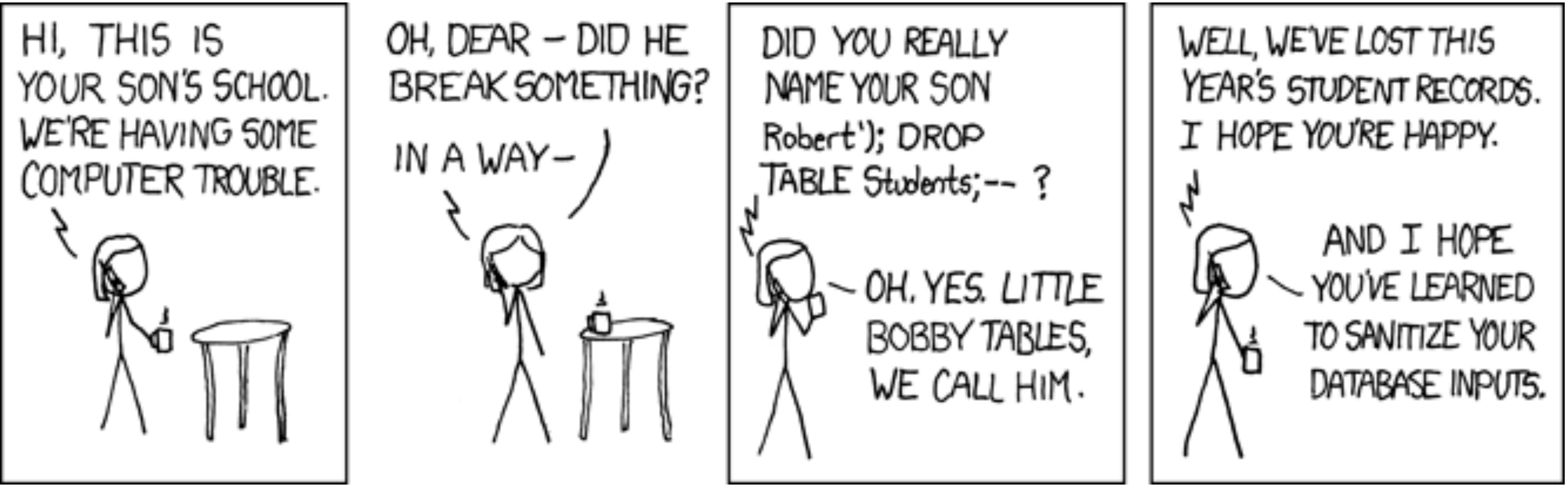
Origin of attacks:

- Remote attacks: attacker remotely breaks into a system, typically over a network
- Local attacks: malicious user gains additional privileges on a machine (usually administrative)

Attacking techniques against a system:

- *Buffer overflow:*
  - Writing data outside of a buffer allocation to influence code execution
- *Password guessing*
  - Attackers gain access through weakly protected accounts
- *Password reuse*
  - Attackers re-use passwords gained from compromised sites
- *Exploiting trust in program input / environment:*
  - It is often possible to maliciously craft input / environment variables to have deleterious side effects
  - Programmers are often unaware of this

System Vulnerabilities – Example for Malicious Inputs


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

https://xkcd.com/327/, Creative Commons Attribution-NonCommercial 2.5 License.

A typical first attack stage: Vulnerability Scans

Scans

- A scan is an active attack to obtain information about a network and its systems. The attacker contacts machines and requests information in a systematic way and analyzes the result.
- Port Scan: scan is to see which ports are open on a machine

Can leak info about

- Network Topology
- Operating System
- Applications and Application Versions
- …

Used to

- Identify potential targets for subsequent attacks

Different types of active scanning

Finding active hosts:

- Network-level scan → Which IP addresses do respond?
- For example, using ICMP Echo Request

Finding active services:

- Transport Protocol Level Scan
- TCP SYN → TCP ACK?

Finding higher level vulnerabilities:

- TLS handshake
- SSH handshake
- HTTP Headers
- SQL injection vulnerabilities

Different types of active scanning

Finding active hosts:

- Network-level scan → Which IP addresses do respond?
- For example, using ICMP Echo Request

Finding active services:

- Transport Protocol Level Scan
- TCP SYN → TCP ACK?

Want to scan the Internet for research? Come talk to me!

Finding higher level vulnerabilities:

- TLS handshake
- SSH handshake
- HTTP Headers
- SQL injection vulnerabilities

Denial of Service attacks

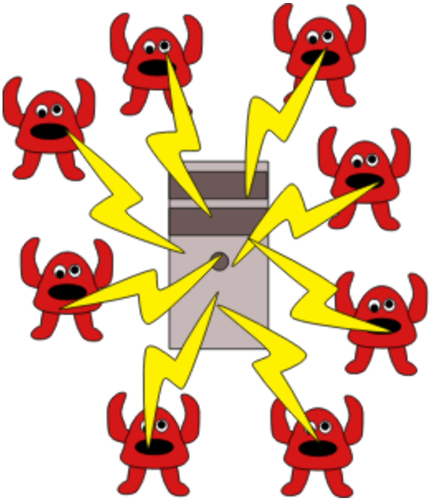
What is Denial of Service?

Denial of Service (DoS) attacks aim at denying or degrading legitimate users' access to a service or network resource, or at bringing down the servers offering such services

Denial of Service Attacking Techniques

Resource destruction (disabling services):

- Gaining access to systems, e.g. through implementation weaknesses as buffer overflow
- Intent to destroy resources, e.g. delete data, shutdown server



Resource depletion by causing:

- Storage of (useless) state information
- High traffic load (requires high overall bandwidth from attacker)
- Expensive computations (“expensive cryptography”!)
- Resource reservations that are never used (e.g. bandwidth)

Origin of malicious traffic:

- Genuine or spoofed source addresses
- Single source (DoS)
- Multiple Sources (DDoS) – for example through a botnet
- Reflection Attacks (DRDoS) – for example using 3rd party NTP/DNS servers

Reflection Attacks

To conduct an reflection attack, 2 characteristics have to be met:

- A service must accept and respond to spoofed packets, typically valid for stateless protocols
- A service’s response packets are larger than the query packets

In an attack, the attacker would send large amounts of packets to this service, with the victim’s address as the source (spoofing). Subsequently, the victim will be flooded with replies from a 3rd party

Typical examples:

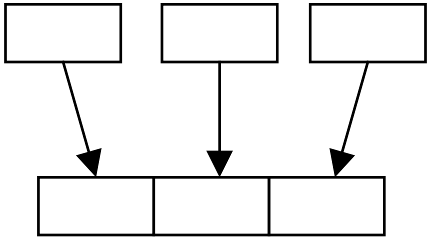
- DNS – very prominent
  - DNS servers are typically well-connected to the Internet
  - DNS is stateless
  - DNS features small queries and large replies
  - Somewhat tractable through rate limiting at the server
- NTP – also commonly used
  - Stateless protocol, some (useless) commands send large replies
  - Easily rate limitable, a NTP server should never use significant amounts of bandwidth

Resource Destruction via protocol edge cases (ancient examples)

Ping-of-Death:

Maximum size of TCP/IP packet is 65536 bytes

Oversized packet may crash, freeze, reboot system

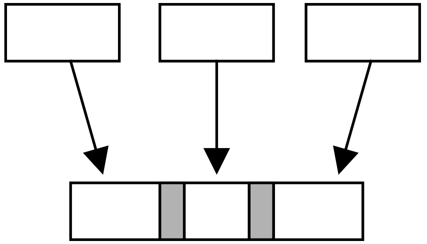


Normal Behavior

Teardrop:

Fragmented packets are reassembled using the Offset field.

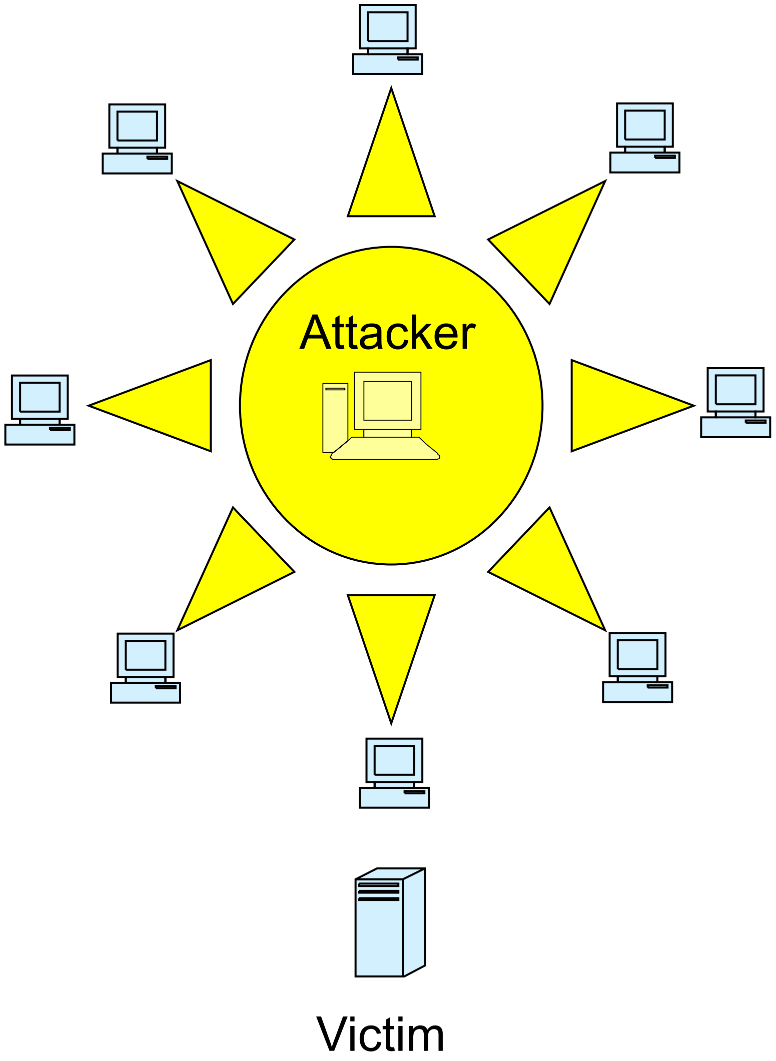
Overlapping Offset fields might cause system to crash.



Teardrop Attack

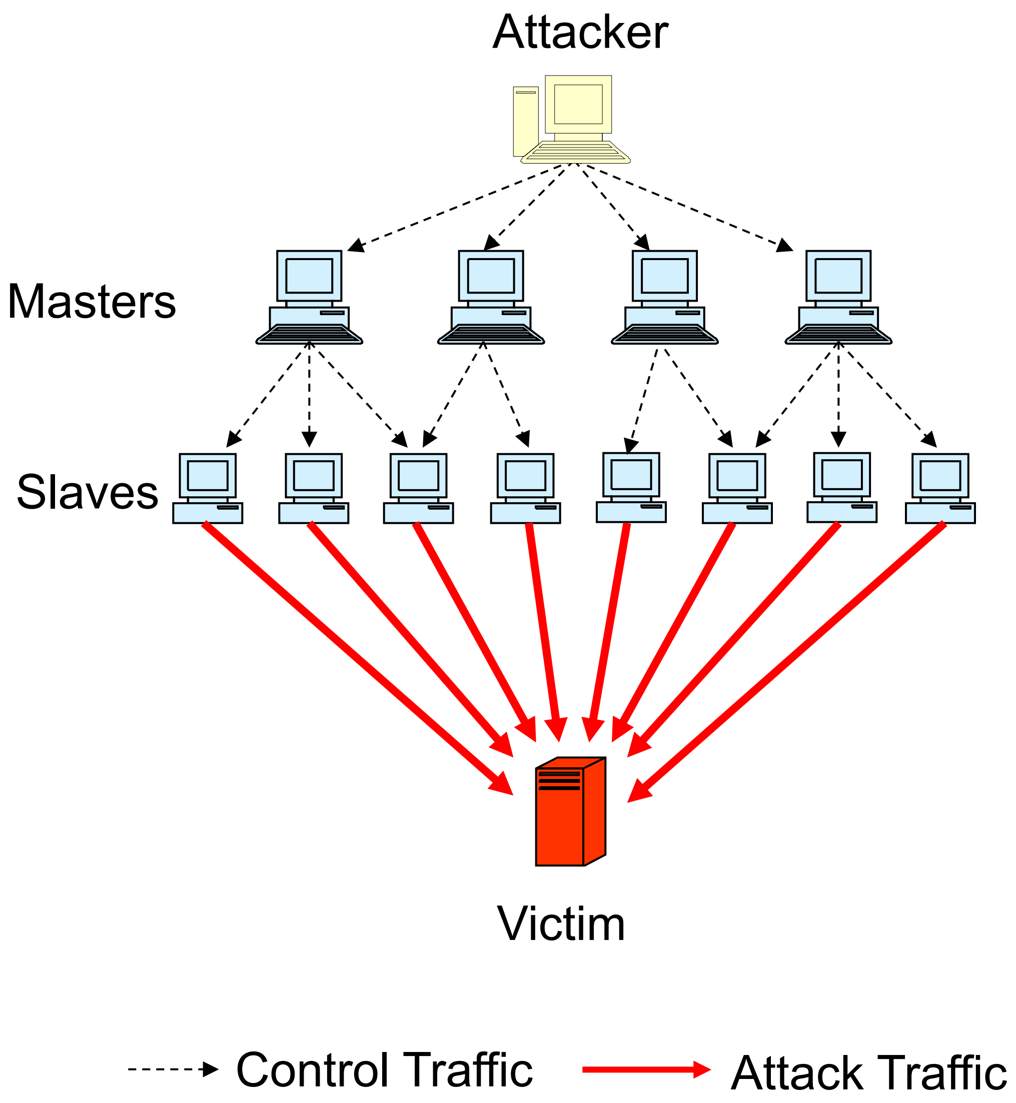
→ In this type of attack, a few packets can be sufficient to bring down a system!

Resource Depletion with Distributed DoS (Stage 1)



- Category *Overwhelming the victim with traffic*
- Attacker intrudes multiple systems by exploiting known flaws
- Attacker installs DoS-software:
  - „Root Kits“ are used to hide the existence of this software
- DoS-software is used for:
  - Exchange of control commands (“Command and Control”, CnC)
  - Launching an attack
  - Coordinating the attack

Resource Depletion with Distributed DoS (Stage 2)

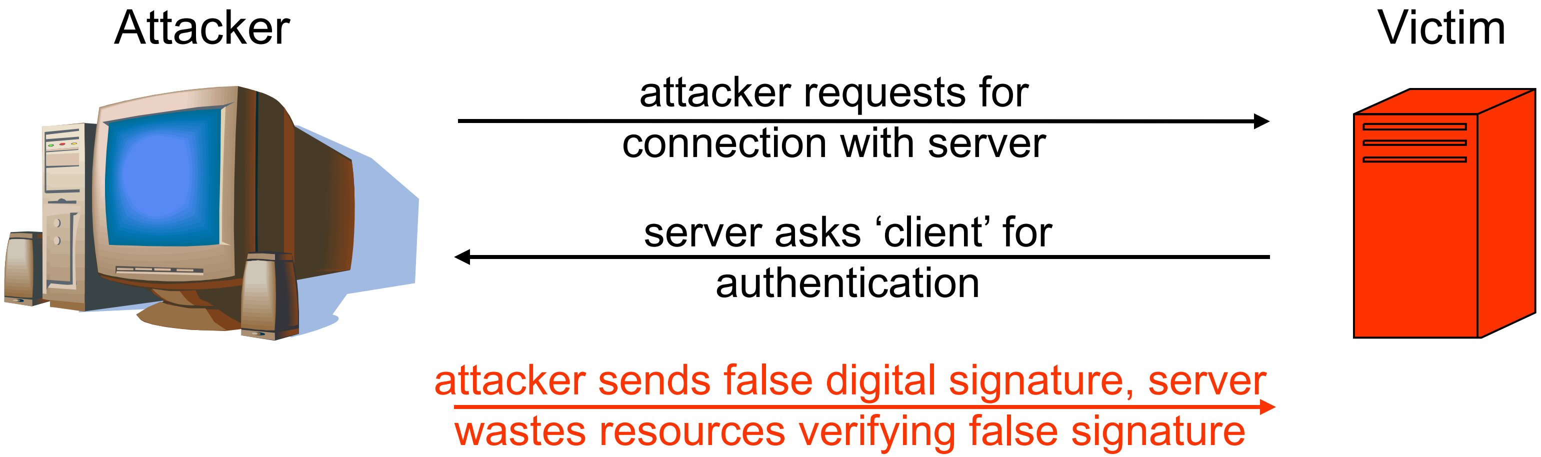


- The attacker classifies the compromised systems in:
  - Master systems
  - Slave systems
- Master systems:
  - Receive command data from attacker
  - Control the slaves
- Slave systems:
  - Launch the proper attack against the victim
- During the attack there is no traffic from the attacker

Resource Depletion with CPU Exhaustion

Category *CPU exhaustion by causing expensive computations:*

Here: attacking with bogus authentication attempts



Attacker

Victim

attacker requests for connection with server

server asks ‘client’ for authentication

attacker sends false digital signature, server wastes resources verifying false signature

- The attacker usually either needs to receive or guess some values of the second message, that have to be included in the third message for the attack to be successful
- Also, the attacker, must trick the victim *repeatedly* to perform the expensive computation in order to cause significant damage

➡ Be aware of DoS-Risks when introducing security functions into protocols!!!

Part I: Attack Prevention

- ❑ Part 0: Attacks
- ❑ Part I: Attack Prevention
- ❑ Part II: Attack Detection
- ❑ Part III: Response Mechanisms
- ❑ Part IV: Recent Attacks and Research

Attack Prevention

Prevention:

All measures taken in order to avert that an attacker succeeds in realizing a threat

Examples:

- Cryptographic measures: encryption, computation of modification detection codes, running authentication protocols, etc.
- Firewall techniques: packet filtering, service proxying, etc.

➡ Preventive measures are by definition taken *before an attack takes place*

Prevention: Defense Techniques Against DoS Attacks (1)

Defenses against disabling services:

- Basic defenses:
  - Good system administration – updates, only needed packets
  - Firewalls, logging & intrusion detection systems
- Implementation weakness defenses:
  - Code reviews, stress testing, fuzzing, etc.
- Protocol deviation defenses:
  - Fault tolerant protocol design
  - Error logging & intrusion detection systems
  - “DoS-aware protocol design”:
    - Do not perform expensive operations, reserve memory, etc., before authentication

Prevention: Defense Techniques Against DoS Attacks (2)

Defenses against resource depletion:

- Generally:
  - Rate Control per user
  - Accounting & Billing (“if it is for free, why not use it excessively?”)
  - Identification and punishment of attackers
- Authentication of clients plays an important role for the above measures
- Memory exhaustion: stateless protocol operation

Concerning origin of malicious traffic:

- Defenses against single source attacks:
  - Disabling of address ranges (helps only if addresses are valid)
- Defenses against forged source addresses:
  - **Ingress Filtering at ISPs** – why can this be effective?
  - **Egress Filtering** (block outgoing packets with source address from other network)
- Widely distributed DoS: ???

Attack Prevention, Detection and Response

❑ Part 0:	Attacks
❑ Part I:	Attack Prevention
❑ **Part II:**	**Attack Detection**
❑ Part III:	Response Mechanisms
❑ Part IV:	Recent Attacks and Research

Part II: Attack Detection

Introduction

Knowledge-based Detection

Anomaly Detection

Introduction

Prevention is not sufficient in practice

What can be attained with intrusion detection?

- Detection of attacks and attackers
- Detection of system misuse (includes misuse by legitimate users)
- Limitation of damage (if response mechanisms exist)
- Deterrence of potential attackers

Introduction (2)

Intrusion

Definition 1

“An Intrusion is unauthorized access to and/or activity in an information system.”

Definition 2 (more general)

“…Any set of actions that attempt to compromise the integrity, confidentiality or availability of a resource.” [HLM91]

As seen in Definition 2, the term “Intrusion” is often used in the literature to characterize any kind of attacks.

Intrusion Detection

All measures taken to recognize an attack *while or after it occurred*

Examples:

Recording and analysis of audit trails

On-the-fly traffic monitoring and intrusion detection.

Why is Intrusion Detection Hard?

Background: Binary Classification

	Attack	**No Attack**
Alarm Raised	True Positive	False Positive
No Alarm Raised	False Negative	True Negative

Source: [Tapiador04]

Example Classifier

	Attack	**No Attack**
Alarm Raised	99.9999%	0.0001%
No Alarm Raised	0.0001%	99.9999%

How would you describe that system?

- Excellent
- Fantastic
- Very good

Example Classifier

	Attack	**No Attack**
Alarm Raised	99.9999%	0.0001%
No Alarm Raised	0.0001%	99.9999%

How would you describe that system?

- Excellent
- Fantastic
- Very good
- ***Total disaster, so sad!***

Example Classifier

	Attack	**No Attack**
Alarm Raised	99.9999%	0.0001%
No Alarm Raised	0.0001%	99.9999%

How would you describe that system?

- Excellent
- Fantastic
- Very good
- ***Total disaster, so sad!***

Assume the Munich Scientific Network handles about 1 millions packets per second (~10 Gbit/s).
This will generate 1 false positive event per second!

Part II: Attack Detection

Introduction

Knowledge-based Detection

Anomaly Detection

Knowledge-based Attack Detection (1)

Idea:

Store signatures of attacks in a database

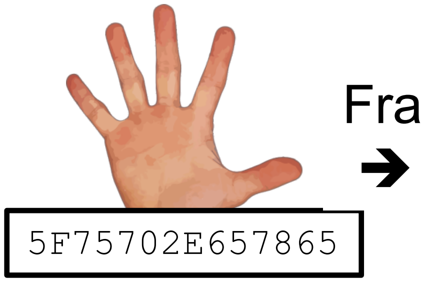
Monitor traffic for signatures

Frequently update signature database

Hand detected
➔ human

Example of a rule in the IDS Snort (http://www.snort.org/)

```
alert tcp $HOME_NET any -> any 9996 \
 (msg:"Sasser ftp script to transfer up.exe"; \
content:"|5F75702E657865|"; depth:250; flags:A+;
classtype: misc-activity; \ sid:1000000; rev:3)
```



Fragment of Sasser located
➔ Sasser

Knowledge-based Attack Detection (2)

Advantages? Drawbacks?

Known attacks can be reliably detected. Very few false positives.

Drawbacks?

Only known attacks can be detected

Slight variations of known attacks are not detected

Knowledge-based Attack Detection (3)

Advantages:

- Known attacks can be reliably detected
- Very few false positives.

Drawbacks:

- Only known attacks can be detected
- Slight variations of known attacks are not detected

Part II: Attack Detection

Introduction

Knowledge-based Detection

Anomaly Detection

Anomaly Detection

Anomaly detection systems include a model “normal” such as:

- normal traffic dynamics
- expected system performance

The current state of the network is compared with this model to detect anomalies.

Alarms can be raised if the current state differs from the “normal“ behavior as defined by this model.

Anomalies can be detected in

Traffic behavior

Protocol behavior

Application behavior

Anomaly Detection – Example and Difficulties

Let‘s look for anomalies in the count of eduroam users:

Average?

Anomaly Detection – Example and Difficulties

Let‘s look for anomalies in the count of eduroam users:

Average: 12.7k

Threshold?

Anomaly Detection – Example and Difficulties

Let‘s look for anomalies in the count of eduroam users:

Average: 12.7k

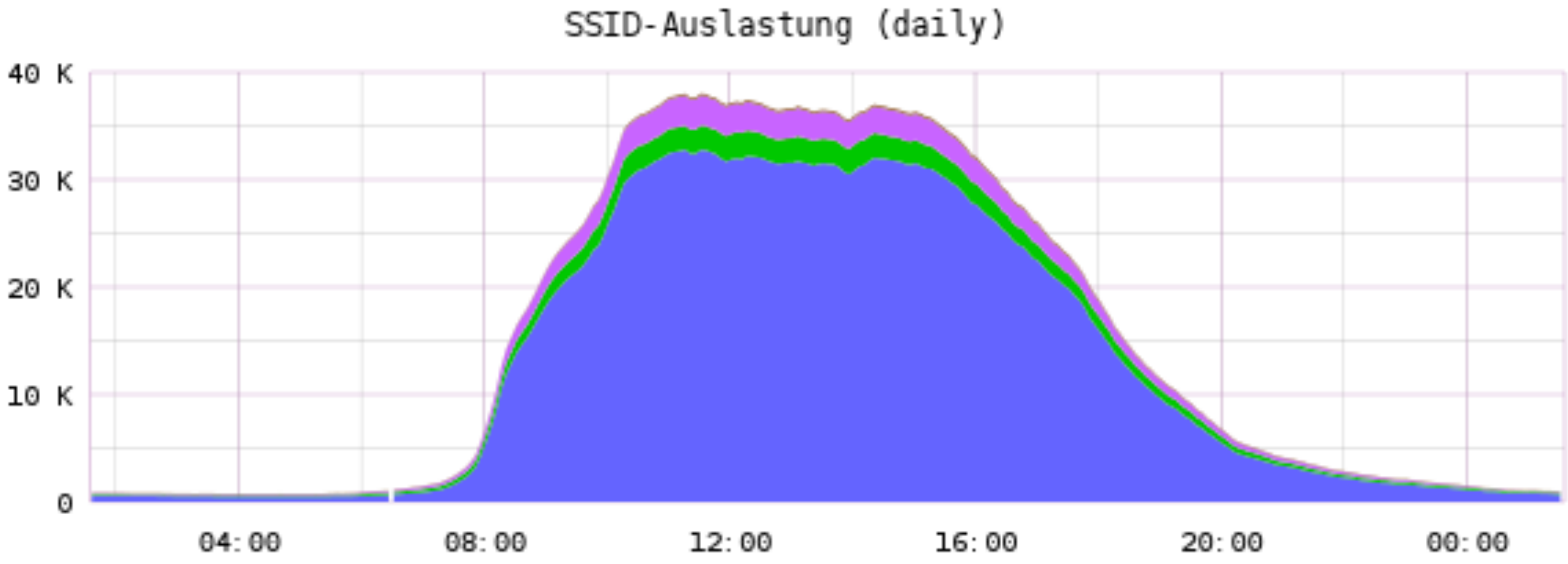
Threshold: 10k < x < 15k

Anomaly Detection – Example

Let‘s look for anomalies in the count of eduroam users:

Average: 12.7k

Threshold: 10k < x < 15k

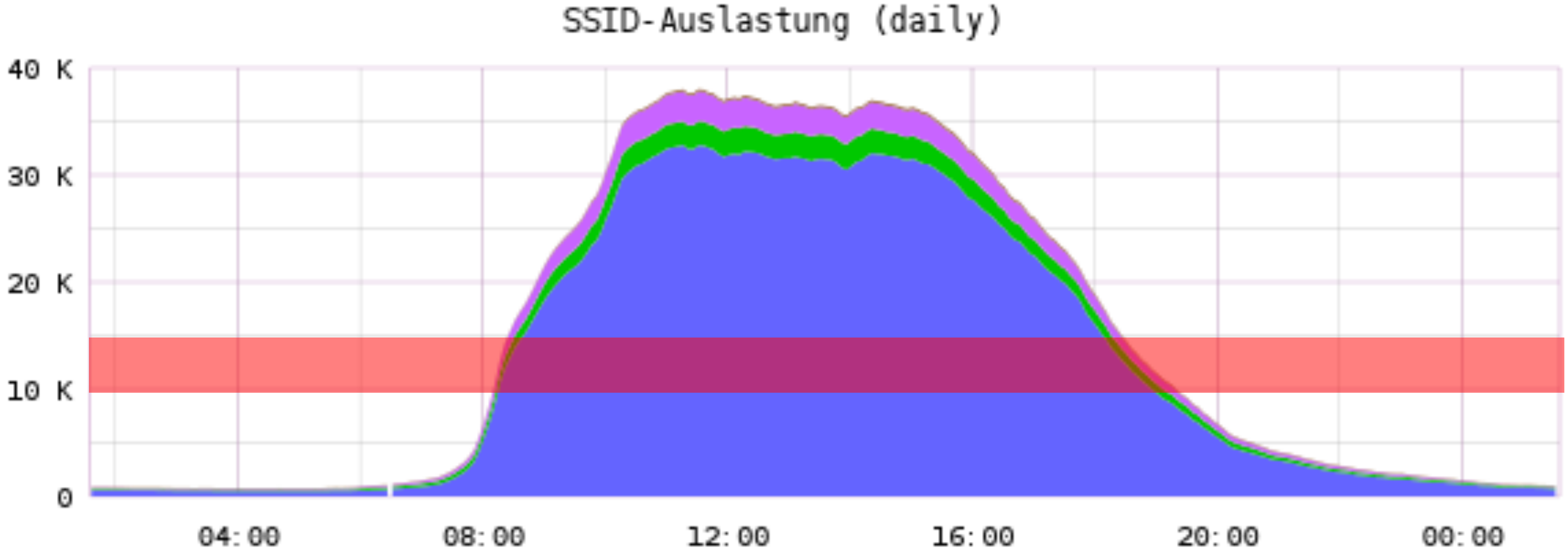


Anomaly Detection – Example

Let‘s look for anomalies in the count of eduroam users:

Average: 12.7k

Threshold: 10k < x < 15k



→ The system is wrong most of the time!

Anomaly Detection – Difficulties

„Diurnal patterns:“ Number of users will fluctuate heavily with:

- Time of day (very few users at night)
- Day of week (very few users on weekend)
- Time of year (fewer users during semester break)
- „Random“ events – public holidays, festivities, ...

Long-term patterns: Number of users will grow over time

- TUM student numbers roughly doubled in the past 10 years

It is very difficult to build a well-functioning anomaly detection system with a very low false positive rate. Real-world use typically limited to raising alerts in narrow edge cases.

In our example, this could be “WiFi users > 50k”, with the highest observed peak so far 42k.

Attack Prevention, Detection and Response

❑	Part 0:	Attacks
❑	Part I:	Attack Prevention
❑	Part II:	Attack Detection
❑	**Part III:**	**Response Mechanisms**
❑	Part IV:	Recent Attacks and Research

Response Strategies

Packet Filtering

- Which packets to filter?
- Packets to offered service (e.g. tcp80) vs other packets (e.g. NTP)?
- At host, at Internet uplink, at ISP?

Rate Limiting

- Congestion control

Tracking

- Traceback techniques

Redirection

Recent Attack Patterns and Research

Recent Attack Patterns and Research

- Part 0: Attacks
- Part I: Attack Prevention
- Part II: Attack Detection
- Part III: Response Mechanisms
- Part IV: Current Examples and Recent Research
  - Understand one of today's typical attack patterns
  - Learn how researchers tackle the problem

Recent Attack Patterns: Password re-use

“My facebook has been hacked”

“Billions of user accounts exfiltrated from company X”

Many recent attacks are based on password re-use:

- Nearly impossible for humans to manually maintain separate passwords for dozens to hundreds services they are using
- This leads to password reuse, which is utilizing the same password for several services
- >40% of users reuse passwords [1]

[1] Das, Anupam, et al. "The tangled web of password reuse." Symposium on Network and Distributed System Security (NDSS). 2014.

Example for password reuse attack:

For some task, you need free software X, provided by EvilCorp.

1. EvilCorp requires you to create an account to download the software.
2. You register as eve@tum.de and use your TUM password to avoid memorizing yet another password.
3. Because EvilCorp deploys low standard security, their unencrypted user database is eventually breached.
4. The attackers now can log into your TUM e-mail addresses to cause more harm:
    - Abusing university resources
    - “Hacking” your facebook account which also uses this e-mail address
    - ….

Can we measure password reuse?

Tripwire: Inferring Internet Site Compromise

Joe DeBlasio, Stefan Savage, Geoffrey M. Voelker, and Alex C. Snoeren *(UC San Diego), Internet Measurement Conference'17*

Step 1: Laying the Bait

- Creating thousands of e-mail addresses at a cooperative e-mail provider
- Use this e-mail to create accounts at thousands of websites, *using the same password as for the email provider*
- Wait …

Tripwire: Inferring Internet Site Compromise,
Joe DeBlasio, Stefan Savage, Geoffrey M. Voelker, and Alex C. Snoeren *(UC San Diego), Internet Measurement Conference'17*

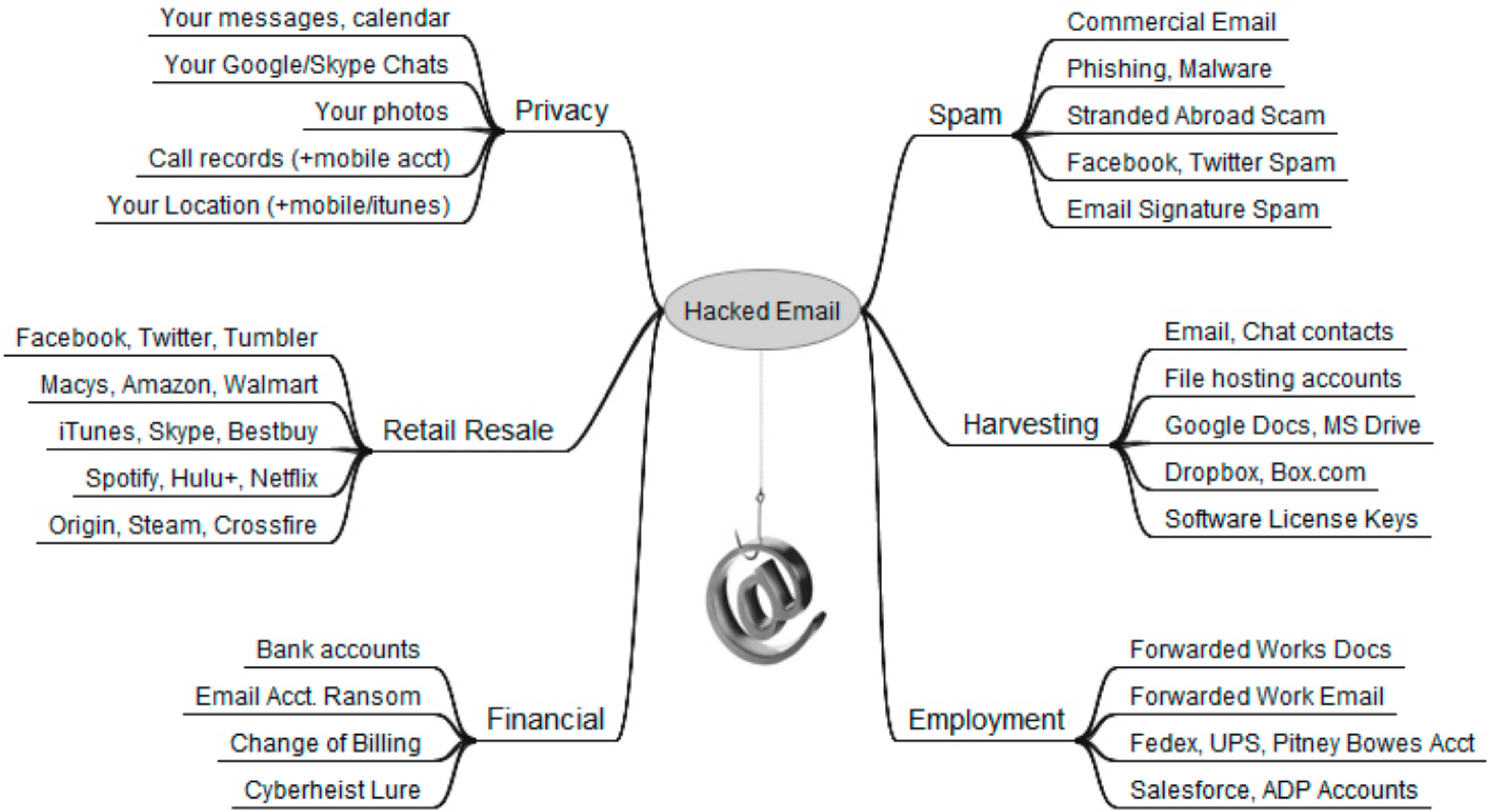
Can we measure password reuse? – Collecting Attackers

Step 2: Collect the attackers

- Closely monitor email accounts for logins
- What would you expect to happen?

Tripwire: Inferring Internet Site Compromise,
Joe DeBlasio, Stefan Savage, Geoffrey M. Voelker, and Alex C. Snoeren *(UC San Diego), Internet Measurement Conference'17*

Can we measure password reuse? – Value and use of compromised email-addresses



https://krebsonsecurity.com/2013/06/the-value-of-a-hacked-email-account/

Can we measure password reuse? – Results

Fresh compromises detected

19 compromises over 24 months; only one previously public

Large and small sites affected

Largest site has ~50m users; >100m users impacted across

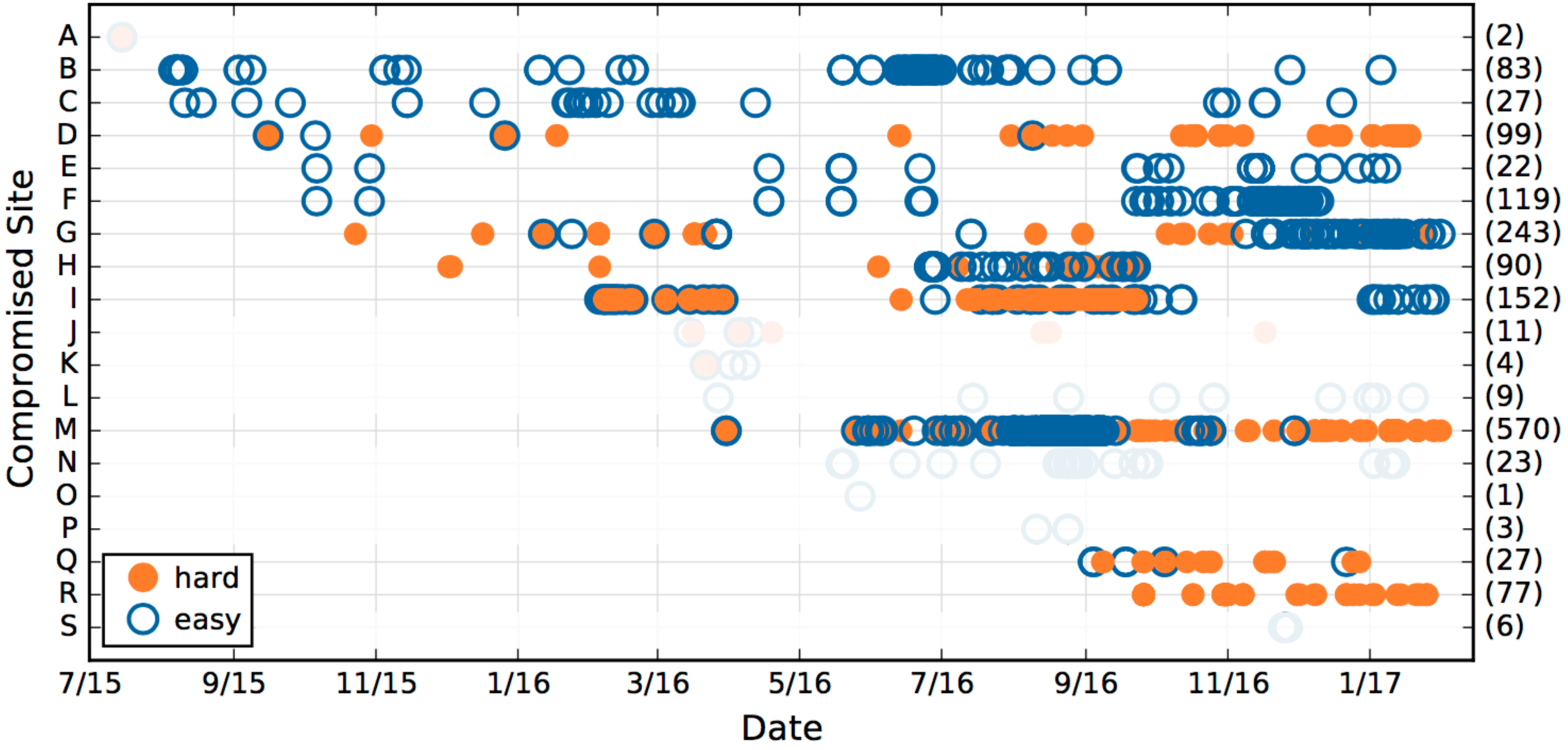
Most accounts are not abused – little indication to user

~25% used for spam; one password changed

~75% of accounts: Attackers waiting for more private data!

Tripwire: Inferring Internet Site Compromise,
Joe DeBlasio, Stefan Savage, Geoffrey M. Voelker, and Alex C. Snoeren *(UC San Diego), Internet Measurement Conference’17*

Can we measure password reuse? – Access Patterns to compromised accounts



DeBlasio et al: https://conferences.sigcomm.org/imc/2017/slides/166-DeBlasio-Tripwire.pdf

Measuring Password Reuse -- Disclosure

Research Team notified breached sites

No site disclosed the breach

More or less strong promises to “investigate”, “fix”

→Sites do not care about your security!

What to do?

Use a password manager

Change your passwords

Setup backup access and 2 Factor Authentication (2FA)

Tripwire: Inferring Internet Site Compromise,
Joe DeBlasio, Stefan Savage, Geoffrey M. Voelker, and Alex C. Snoeren *(UC San Diego), Internet Measurement Conference’17*

The End

References

[HLM91] Heberlein, Levitt und Mukherjeeh. A method to detect intrusive activity in a networked environment. In Proceedings of the 14th National Computer Security Conference, 1991.

[Mirkovic2004] J. Mirkovic and P. Reiher, "A Taxonomy of DDoS Attack and DDoS Defense Mechanisms," *ACM SIGCOMM Computer Communication Review*, vol. 34, April 2004, pp. 39-53.

[Tapidor2004] J. M. Estevez-Tapiador, P. Garcia-Teodoro, and J. E. Diaz-Verdejo, "Anomaly detection methods in wired networks: a survey and taxonomy," Computer Communications, vol. 27, July 2004, pp. 1569-1584.