Recap

- Service-level agreement (SLA)
- Monitoring
- Logging
- Post-mortem analysis
- Load balancing
- Scaling
  - Monitoring of scaling

Goals of LSD

- Train the student to develop large-scale IT systems, where scalability is a key characteristic
- The student must have knowledge of concepts, techniques and technologies for the continuous integration and delivery of software-based systems
- The student must be able to design, implement, and maintain large distributed systems in distributed development teams

See also: Curriculum 2017 (pdf)

Goals of the DevOps part

- Give you theoretical and practical knowledge on maintening and operating large systems

1. Monitoring — 5. November
2. Logging — 12. November
3. Scaling — 19. November
4. Security — 26. November

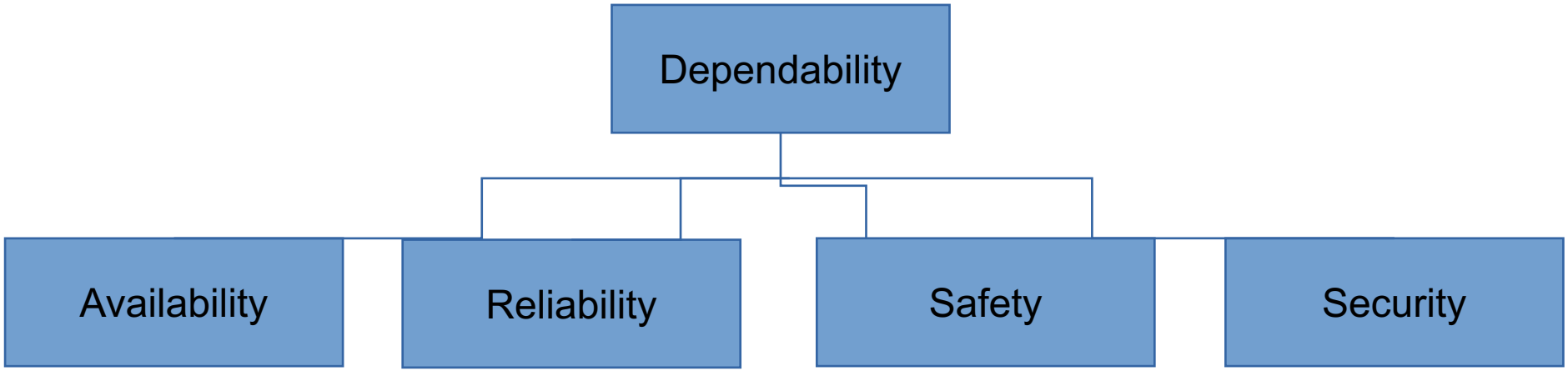
- Essentially everything that happens *around* the code

Goals for today

- Understand what a critical system is
- Understand and apply threat modeling
- Understand and apply risk matrices
- Gain practical knowledge on finding and mitigating breaches
- Gain practical knowledge on intrusion detection

- Literature: DevOps introduction

Dependability



See also: Ian Sommerville: Software Engineering

Laws and regulations

- Regulation (EC) No 45/2001 on the protection of individuals with regard to the processing of personal data
- Directive 95/46/EC on the protection of individuals with regard to the processing of personal data and on the free movement of such data
- Commission Decision 2001/497/EC of 15 June 2001 on standard contractual clauses for the transfer of personal data to third countries
- International Safe Harbor Privacy Principles
- Directive 2002/58/EC of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector
- Directive 2006/24/EC of 15 March 2006 on the retention of data generated or processed in connection with the provision of publicly available electronic communications services or of public communications networks
- Council Directive 2008/114/EC of 8 December 2008 on the identification and designation of European critical infrastructures
- Council Framework Decision 2005/222/JHA of 24 February 2005 on attacks against information systems.

Attacker types

- Script kiddies
  - Low threat, low profile
- Black hat groups
  - Hight threat, high profile
- Government groups
  - Hight threat, political profile
- White hats
  - Low threat, political profile

Security

- Secure from what?
  - The who/where
- What are you protecting?
  - The what
- When are you secure?
  - The how

Threats

- A threat is a combination of
  - Intent
  - Capability
  - Opportunity

Threats

- Intent
  - Hard to do anything about, but don’t be idiots
- Capability
  - Impossible to change
- Opportunity
  - This is our focus

Intelligence

(Not the “I’m smart” intelligence)

- Knowledge of attackers to protect from
  - Actionable
  - On a strategic, operational, tactical level

Intelligence

- Strategical level
  - Broad issues of business values, economy, political
- Operational level
  - Design of practical countermeasures and policies
- Tactical
  - Practical level: information about current threats and priorities

Intelligence tasking

- Knowledge of attackers to protect from
  - Actionable
  - On a strategic, operational, tactical level
- Tasking
  1. Collect
  2. Analyze
  3. Process
  4. Disseminate

Intelligence tasking

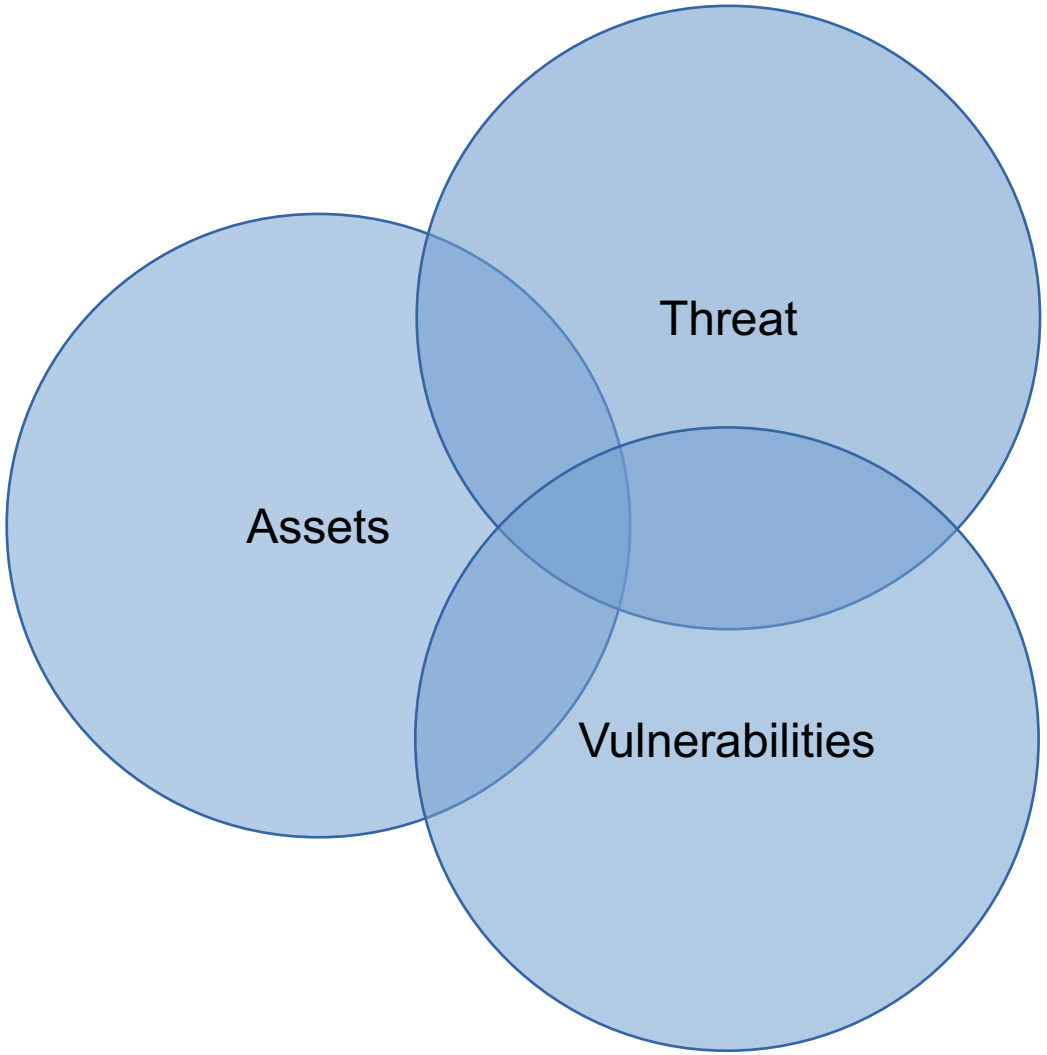
1. Collect
    - Gather information
    - What are your assets? What is worth protecting?
2. Analyze
    - Analyze adversary and opportunities
    - What are the threats and vulnerabilities?
3. Process
    - Process the information so far
    - What are the risks? Which risks are worth protecting from?
4. Disseminate
    - Decide and implement mitigations

Threat modelling

- Threats
  - The who
  - Who/what is the threat and what can they do?
- Assets
  - The what
  - What are you trying to protect?
- Vulnerabilities
  - The how
  - Where are you vulnerable?
  - Attack vectors

Threat modelling

- Threats
- Assets
- Vulnerabilities



Intelligence tasking

1. Collect
    - Gather information
    - What are your assets? What is worth protecting?
2. Analyse
    - Analyse adversary and opportunities
    - What are the threats and vulnerabilities?
3. Process
    - Process the information so far
    - What are the risks? Which risks are worth protecting from?
4. Disseminate
    - Decide and implement mitigations

Critical systems

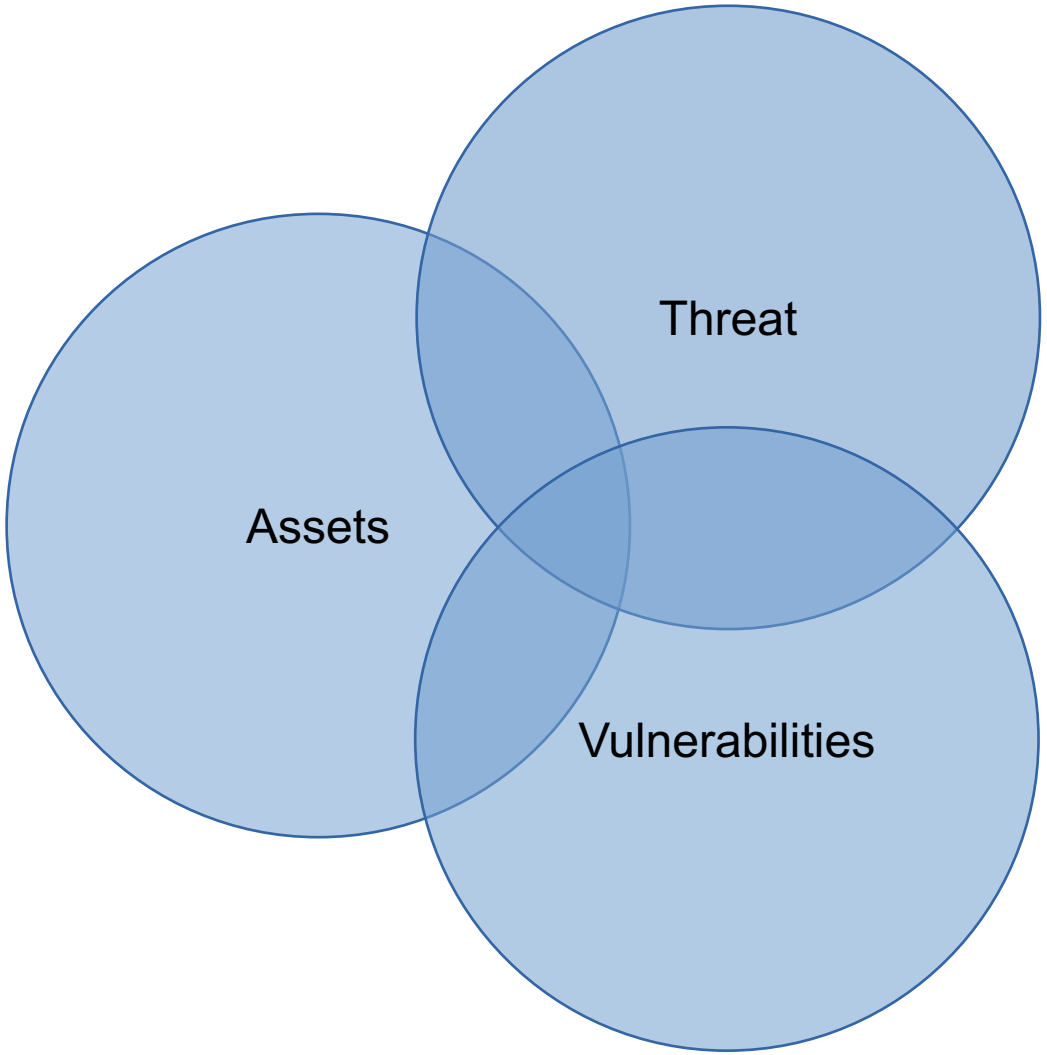
"a system which must be highly reliable to avoid incurring prohibitive costs" - Wikipedia

- Safety critical
  - Failure leads to death
- Mission critical
  - Failure may lead to death
- Business critical
  - Failure leads to economic loss
- Security critical
  - Failure leads to data loss

Threat modelling

- Threats
- Assets
- Vulnerabilities

- Safety critical
- Mission critical
- Business critical
- Security critical



Risk matrices

- All this is about risk
  - How do we assess risks?
- Severity
  - Catastrophic – Multiple Deaths
  - Critical – One Death or Multiple Severe Injuries
  - Marginal – One Severe Injury or Multiple Minor Injuries
  - Negligible – One Minor Injury

Risk matrices

- All this is about risk
  - How do we assess risks?
- Likelihood
  - Certain
  - Likely
  - Possible
  - Unlikely
  - Rare

Risk matrices

- All this is about risk
  - How do we assess risks?

	Negligible	Marginal	Critical	Catastrophic
Certain	High	High	Extreme	Extreme
Likely	Moderate	High	High	Extreme
Possible	Low	Moderate	High	Extreme
Unlikely	Low	Low	Moderate	Extreme
Rare	Low	Low	Moderate	High

Cyber threat matrix

- Variant of the risk matrix

Table 1. Generic threat matrix

Threat Level	THREAT PROFILE						
	Commitment			Resources			
					Knowledge		
	Intensity	Stealth	Time	Technical personnel	Cyber	Kinetic	Access
1	H	H	Years to decades	Hundreds	H	H	H
2	H	H	Years to decades	Tens of tens	M	H	M
3	H	H	Months to years	Tens of tens	H	M	M
4	M	H	Weeks to months	Tens	H	M	M
5	H	M	Weeks to months	Tens	M	M	M
6	M	M	Weeks to months	Ones	M	M	L
7	M	M	Months to years	Tens	L	L	L
8	L	L	Days to weeks	Ones	L	L	L

Reproduced from Duggan et al. [8].

Intelligence tasking

1. Collect
    - Gather information
    - What are your assets? What is worth protecting?
2. Analyse
    - Analyse adversary and opportunities
    - What are the threats and vulnerabilities?
3. Process
    - Process the information so far
    - What are the risks? Which risks are worth protecting from?
4. Disseminate
    - Decide and implement mitigations

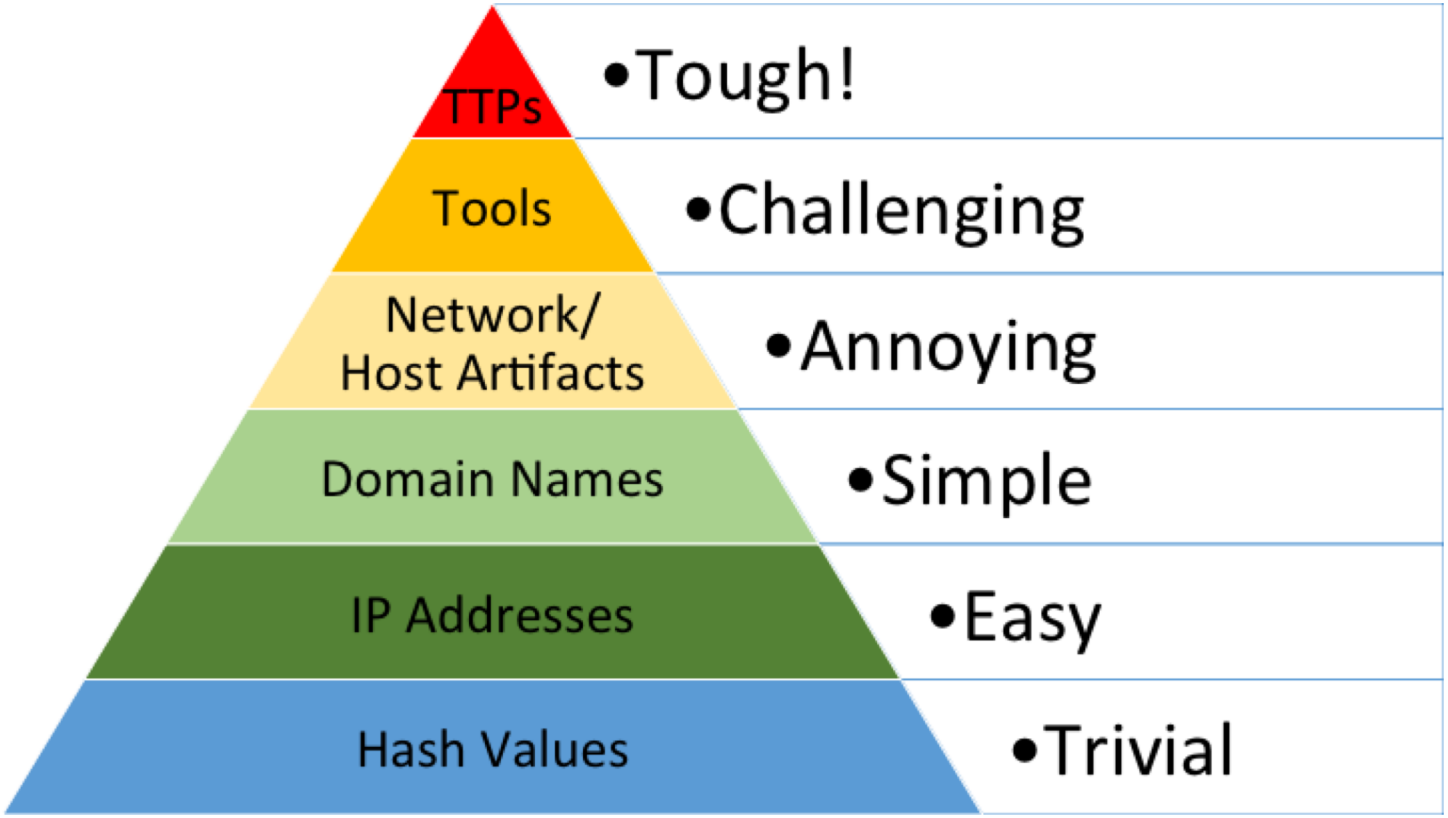
The pyramid of pain

- How can you actually detect intrusions?

- Hash values
- IP addresses
- Domains
- Network/host artifcats
- Tools
- Tactics, Techniques and Procedures

The pyramid of pain

- How can you actually detect intrusions?



See also: David Bianco on the pyramid of pain

Penetration testing

- Open Web Testing Framework
- Automates part of pentesting
- https://owtf.github.io/

OWASP

- Open Web Application Security Project

OWASP Top 10 - 2013	→	OWASP Top 10 - 2017
A1 – Injection	→	A1:2017-Injection
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	U	A4:2017-XML External Entities (XXE) [NEW]
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	U	A7:2017-Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]

Intrusion kill chain

- Another military term
  - Identify, dispatch, decide, attack and resolve

See also: Kill chain on Wikipedia

Phases of the Intrusion Kill Chain

Intrusion kill chain

- Detect: determine whether an attacker is poking around
- Deny: prevent information disclosure and unauthorized access
- Disrupt: stop or change outbound traffic (to attacker)
- Degrade: counter-attack command and control
- Deceive: interfere with command and control
- Contain: network segmentation changes

See also: Kill chain on Wikipedia

Intrusion detection

- Finding out that you are actually under attack!
  - It's hard. Sorry!

1. Develop a baseline for "normal"
   - Traffic, logins, elevation etc.
2. Stop intruders from taking information *out*
   - Firewalls, traffic filtering, white/black listing
3. Train personnel

See also: 3 steps for intrusion detection

Penetration testing (pentesting)

- Just like with software you can test security
- Simulated attacks on your systems
- Requires you to know potential vulnerabilities

See also: Penetration testing on Wikipedia, Kali linux

OWTF

- Open Web Testing Framework
- Automates part of pentesting
- https://owtf.github.io/

Next hand-in

Deadline: **2nd of December 23:59:55**

1. Define your assets
2. Create a risk matrix of your project
3. As operators:
    - Try to find at least one vulnerability in the project you are operating
    - Run OWTF or take one of the OWASP top 10
    - Try to find the attack in the logs

Hand-in: Report containing the above