Infrastructure Design For The Professionally Paranoid

Or: Ticking The Boxes For Fun And Profit

SMARKETS

Mika Boström <mika.bostrom@smarkets.com>

Infrastructure Engineer, Information Security Officer[™], semi-professional interviewer, Systems Wizard

This is a non-presentation version, suitable for distribution

Quick Betting Exchange Overview

- Trading exchange, for sports bets
  - Exchange Core: erlang
  - Most everything else: Python
- All the technical challenges of an investment bank
  - Without the neckties
- Smarkets founded in 2008, now >50 employees
  - More than 30 in engineering
- *We facilitate gambling*
- Regulated as a gambling company, operates as a FinTech company
- Latency is king, transactional integrity is everything
- Industry's traditional reputation is a **BIG** factor

Some Technology Details

- Exchange - Erlang
- Exchange communication channels - Erlang
- Frontends - Python
- All in-house services - Python
- Infrastructure Tooling - Python
- Glue - [*VARIOUS*]
- Production covers >120 nodes
- Peak traffic - Grand National, 425Mb (excluding page loads!)

Audits Are Good Thing (Really!)

- ISO 27001
- Only sounds unappealing
- Encourages to do things sensibly

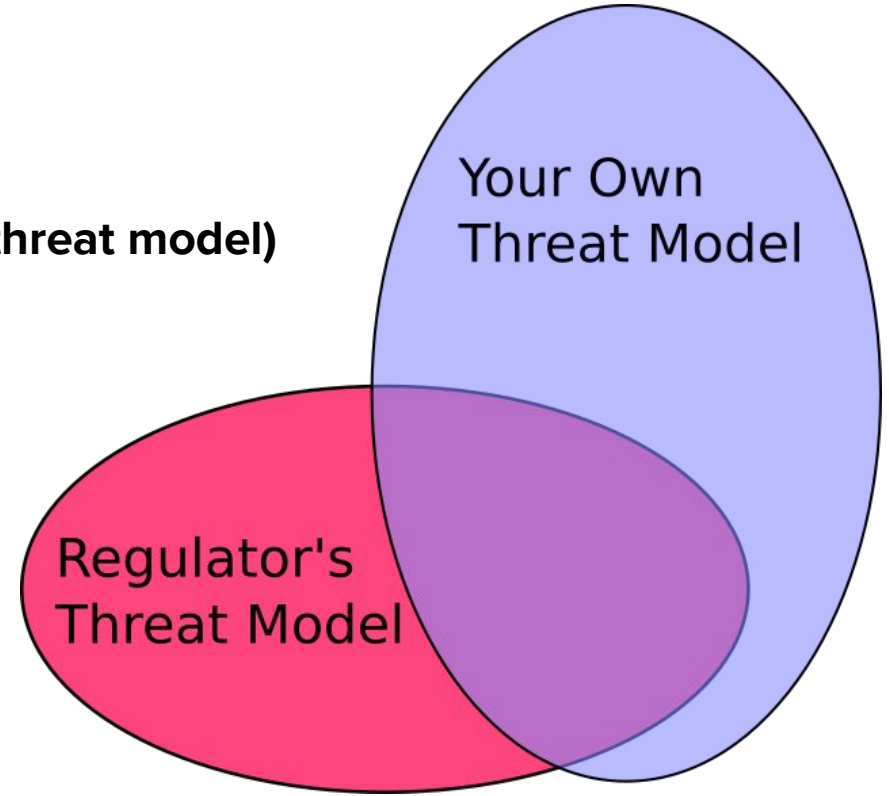
Information Security 001

Everything starts with a question:

What is your threat model?

For Businesses In Regulated Industries...

union(regulator's threat model, own threat model)



Your Own Threat Model

- System breach or break-in
- Data loss
- DDoS
- Customer data disclosure
- Website security
- Unauthorized system access
- Malware
- Weak and/or reused passwords
- ....

Your Own Threat Model	Mitigations
System breach or break-in	detection systems
Data loss	backups, redundancy
DDoS	professional shield
Customer data disclosure	user training, system & comms security
Website security	development best practices, cryptography
Unauthorized system access	unprivileged accounts
Malware	training, system security
Weak and/or reused passwords	password manager, high-entropy pws
....	....

Regulator’s Threat Model - Real Questions Asked

- *Who can make changes to code?*
- *Who can make a release of new code?*
- *How are customer details stored?*
- *How are communications protected?*
- *Who have access to production systems?*
- ***How do you ensure confidentiality?***
- ***Who controls the hardware?***
- ***Do you really expect us to trust the cloud?!***

Regulator's Threat Model

- *Who can make changes to code?*
- *Who can make a release of new code?*
- *How are customer details stored?*
- *How are communications protected?*
- *Who have access to production systems?*
- ***How do you ensure confidentiality?***
- ***Who controls the hardware?***
- ***Do you really expect us to trust the cloud?!***

Real Answers

Anyone

Anyone

Encrypted

Encrypted, Isolated

Engineers

Encryption, isolation

Cloud Provider

Yes, here's why: ...

Gambling Regulator’s Underlying Fear, Distilled

“If you suddenly run off with the customer funds, how do we make sure we can reconstruct the balances and pay everyone what they are owed?”

Technological Choices Are Affected

- The regulatory body must understand the architecture
- The regulatory body must **approve** the architecture
- The regulatory body must have confidence that we can rebuild the entire system in another environment
- *... fast*

Just in case Amazon goes out of business, you know...

Net Result

- Many of the low-hanging fruits in Cloud Best Practices become questionable
- Data breaches are a real threat
- For all practical purposes, cloud equals use of virtual machines
  - *“Who else has access to hardware?”* is not a theoretical problem
  - Cross-VM attacks to extract encryption keys are feasible [1,2]
- No control over media decommissioning
- Securing cross-system communications is important
  - *“What data could be extracted by dumping traffic?”*

1: https://www.cs.unc.edu/~reiter/papers/2012/CCS.pdf
2: https://eprint.iacr.org/2014/435.pdf

Ticking The Boxes

- ❏ Eliminate cross-VM attack vector
- ❏ Data leak from media disposal
- ❏ In-transit data snooping
- ❏ Traffic encryption, authentication
- ❏ System access
- ❏ Admin rights
- ❏ Rapid code changes
- ❏ Infrastructure changes
- ❏ Reproducible accounting, seizable hardware

- ❏ Dedicated tenancy
- ❏ Store all critical data encrypted
- ❏ **TLS everywhere**
- ❏ **Private Certificate Authority**
- ❏ SSH key logins only
- ❏ Principle of least privilege
- ❏ **Mandatory code reviews**
- ❏ Treat configuration as code
- ❏ Up-to-date offsite backups in regulator’s jurisdiction

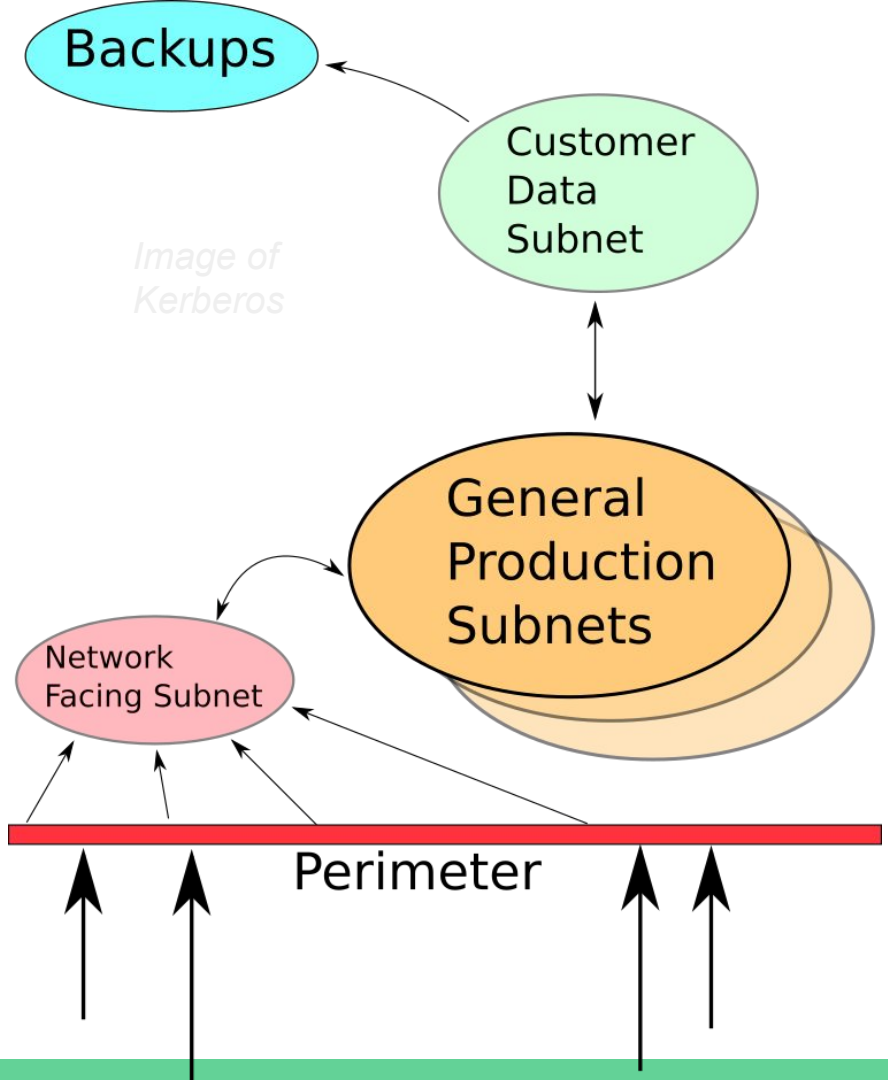
Keep It Simple

- If it contains customers' personally identifiable information, store on encrypted volumes
- … on a separate, locked-down network
- … where all virtual hosts are on dedicated tenancy systems

Logically very close to colo-hosted, owned hardware

Just in case Amazon screws up with their media disposal...

It Can Look Pretty, Too



Databases - The Bonus Sector

- Replicate all production databases
- - Both as local read replicas (to spread the load)
- - And as remote off-site copies
- Take weekly full snapshots
- 3-2-1 rule for backups: 3 copies, 2 formats, 1 off-site

- Best part: *disaster recovery steps for a database are identical to spinning up a read replica*

The TL;DR Version

- Exchange is a complex beast
- Regulators are slow to adapt, but often reasonable
- Just trying to tick boxes is counter-productive
  - Find ways to make things easier to maintain
- Regulators' threat models are different from individual companies'
- Concept of shared resources makes gambling regulators balk
  - Not having control of storage media is scary
- Disaster recovery planning involves PR for two parties

We’re hiring!

https://smarkets.com/careers

Come build the future with us. Pipeline highlights:

- Completely hands-free autoscaling
- Handle 100k concurrent real-time connections (WIP)
- React(ive) frontends
- Kubernetes (!)
  - *(Needs some[™] build/deployment refactoring)*

Mika Boström <mika.bostrom@smarkets.com>