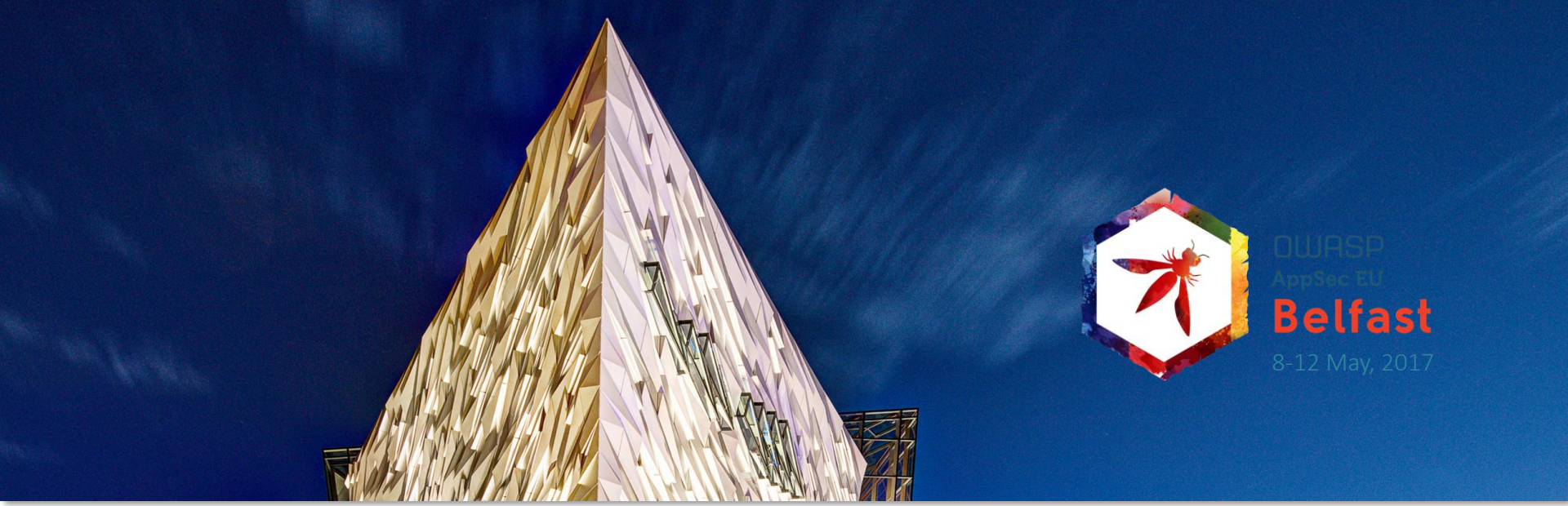
Incremental Threat Modelling

Irene Michlin
Principal Security Consultant



Never try to boil an ocean

Who am I

- Coming from software development and architecture
  - 20 years as software engineer, architect, technical lead
- Variety of consulting and testing work
  - From corporations to start-ups
- Favourite engagement type – threat modelling

Agenda

- STRIDE – quick recap
- Introducing our example
- Incremental modelling walk-through
- Sting in the tail
- Conclusions
- Q&A

Threat modelling - reminder

- Decompose architecture using DFDs
- Search for threats using STRIDE
- Rank or quantify – out of scope for today

Data Flow Diagrams

External Entity	Process	Data Flow	Data Store	Trust Boundary
• People • Other systems	• Logical component • Service • Process in memory	• RPC • Network traffic • File I/O	• Database • File • Queue/Stack	• Process boundary • Network boundary





STRIDE

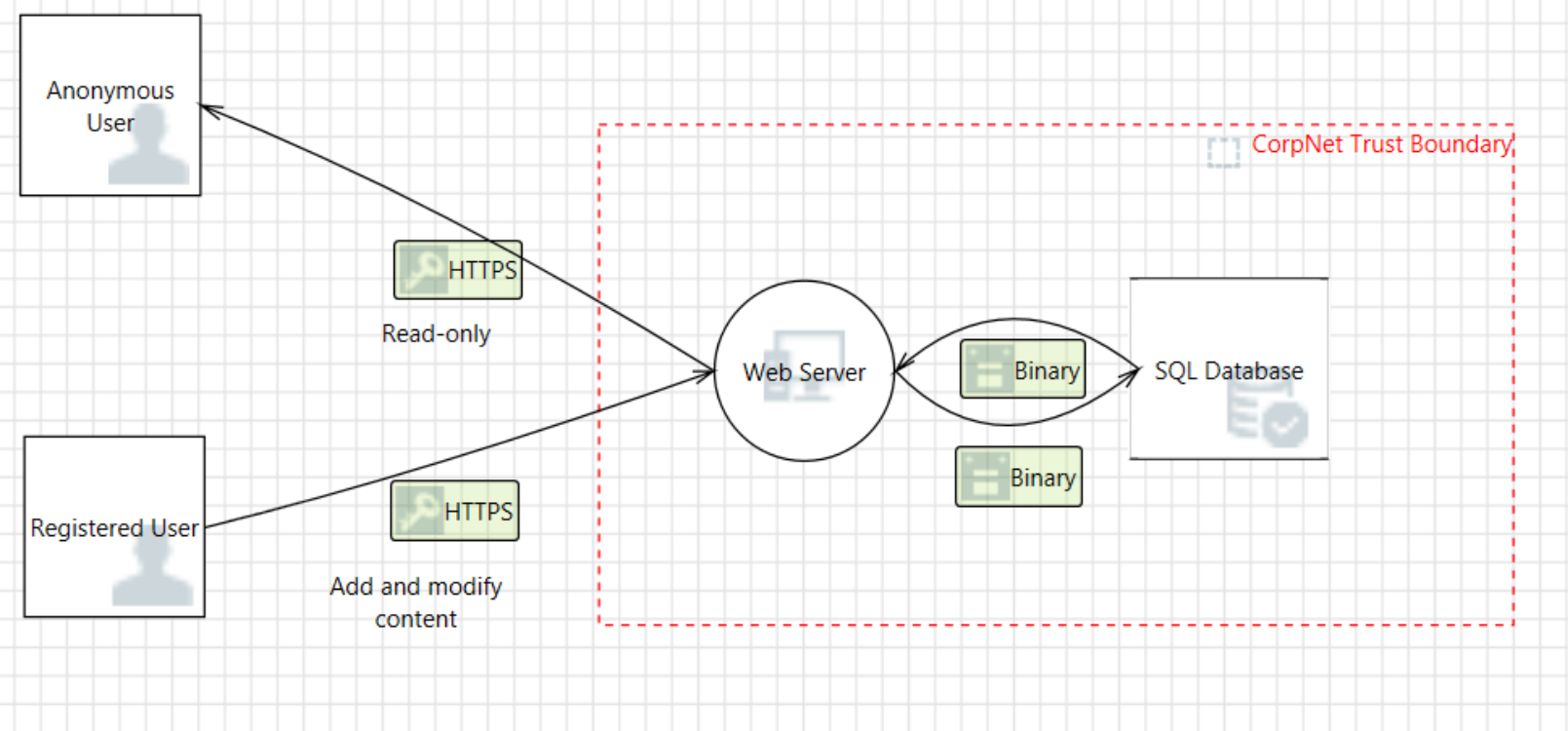
OWASP AppSec EU Belfast

Threat	Property	Definition
Spoofing	Authentication	Impersonating something or someone else
Tampering	Integrity	Modifying data or code
Repudiation	Non-repudiation	Claiming to have not performed an action
Information Disclosure	Confidentiality	Exposing information to non-authorised party
Denial of Service	Availability	Deny or degrade service
Elevation of Privilege	Authorization	Gain capabilities without proper authorisation

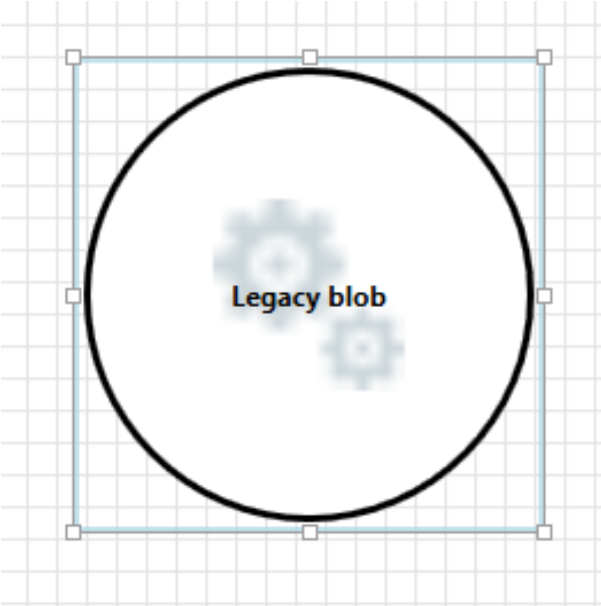
Introducing our example

- Explain the existing architecture and the feature we are planning to add
- Pretend that threat model for the existing part does not exist
- Model new feature

A very simple architecture

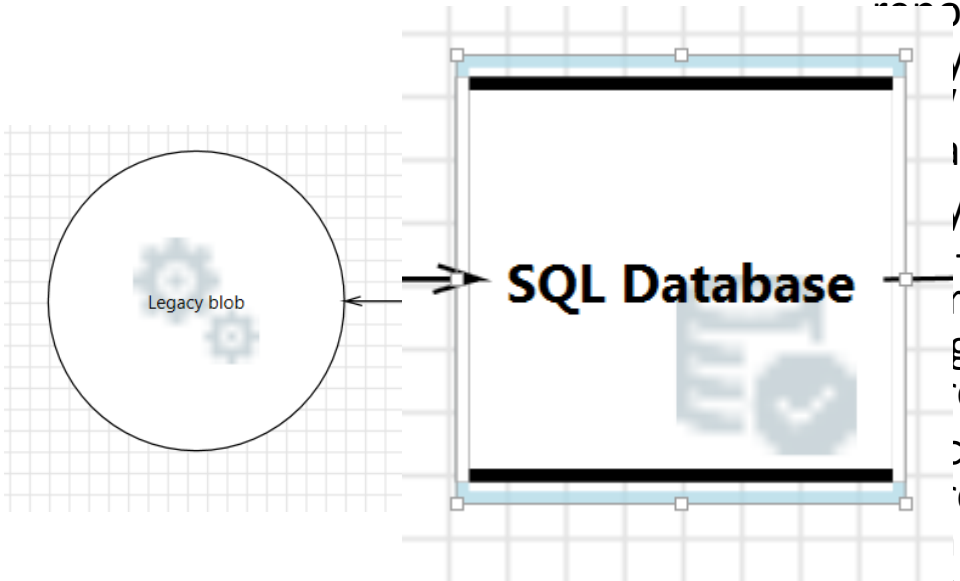


Now pretend to forget it



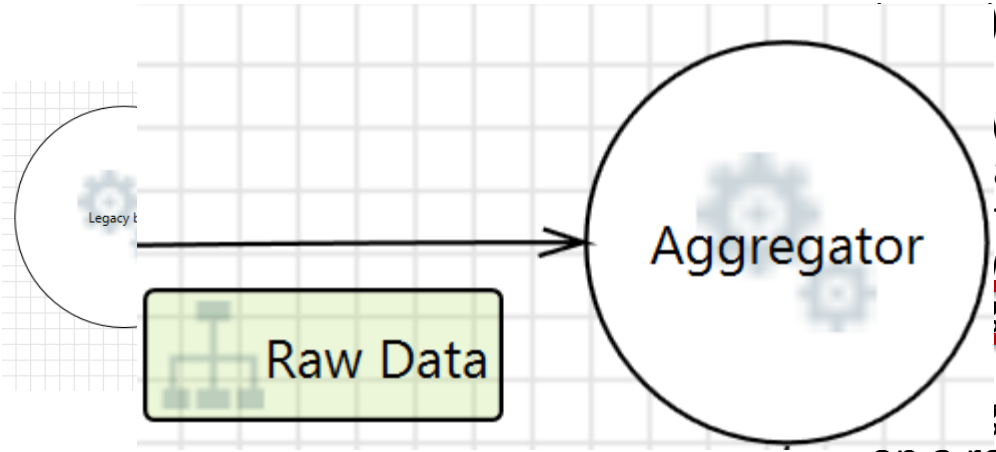
We are going to use a 3rd party reporting and analytics technology. They are going to host Data Warehouse (DWH) and reporting server on their infrastructure.
They will give us licences to use their web-based Analytics App, which can query the reporting server.
The only thing we need to build in-house is an aggregator process,
which will get data from our database, aggregate it and upload it to the DWH on a regular basis (they provide API for automated upload).

Step by step



We are going to use a 3rd party
orting and analytics technology.
are going to host Data Warehouse
H) and reporting server on their
structure.
will give us licences to use their
-based Analytics App, which can
y the reporting server. The only
g we need to build in-house is an
egator process,
h will get data from our database
egate it and upload it to the DWH
regular basis (they provide API for
automated upload).

Step by step

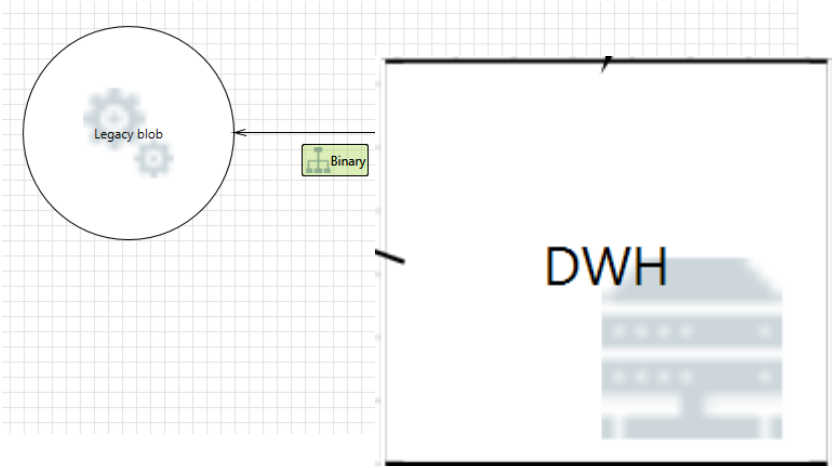


We are going to use a 3rd party reporting and analytics technology. They are going to host Data Warehouse) and reporting server on their ructure.

vill give us licences to use their ased Analytics App, which can the reporting server. The only ve need to build in-house is an gator process,

will get data from our database, gate it and upload it to the DWH on a regular basis (they provide API for automated upload).

Step by step

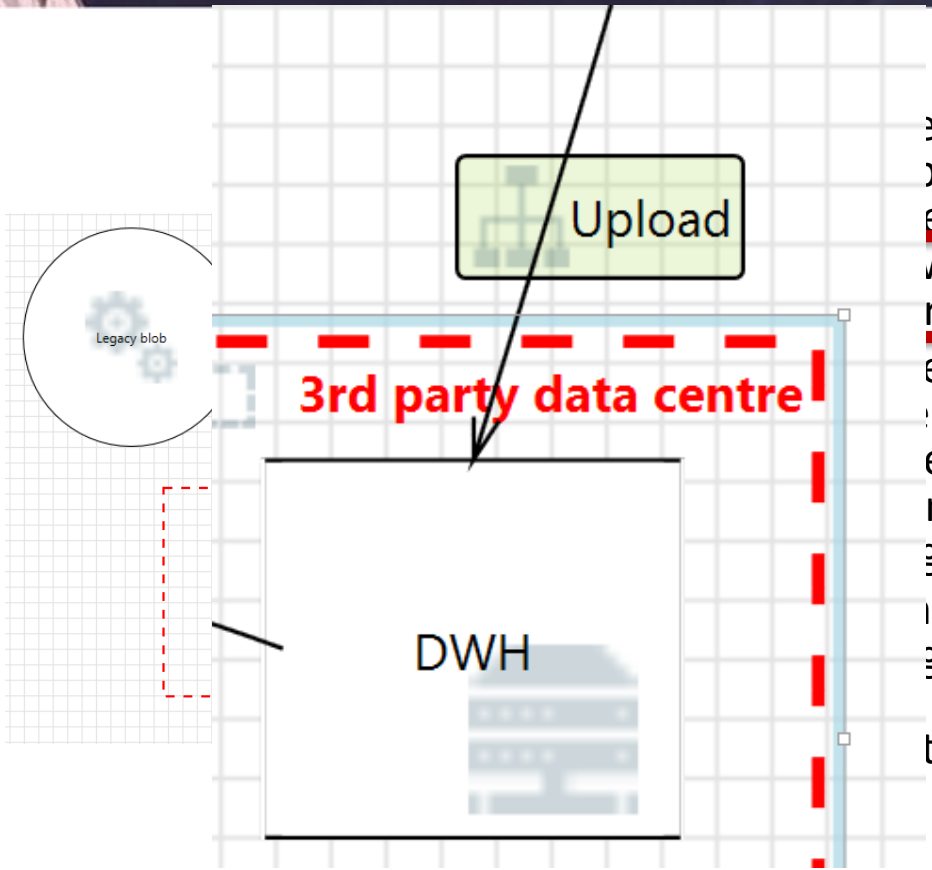


We are going to use a 3rd party reporting and analytics technology. They are going to host Data Warehouse (DWH) and reporting server on their infrastructure.

They will give us licences to use their web-based Analytics App, which can query the reporting server. The only thing we need to build in-house is an aggregator process,

which will get data from our database, aggregate it and upload it to the DWH on a regular basis (they provide API for automated upload).

Step by step

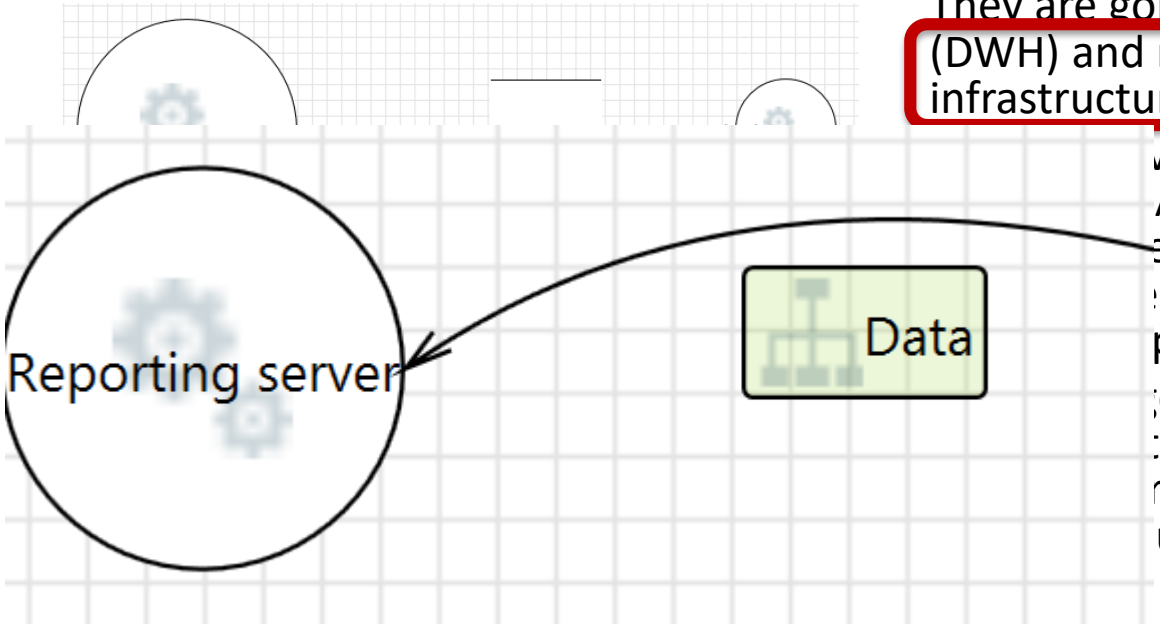


e are going to use a 3rd party
porting and analytics technology.
ey are going to host Data Warehouse
WH) and reporting server on their
rastructure.

ey will give us licences to use their
b-based Analytics App, which can
ery the reporting server. The only
ng we need to build in-house is an
gregator process,

ich will get data from our database,
gregate it and upload it to the DWH
a regular basis (they provide API for
tomated upload).

Step by step

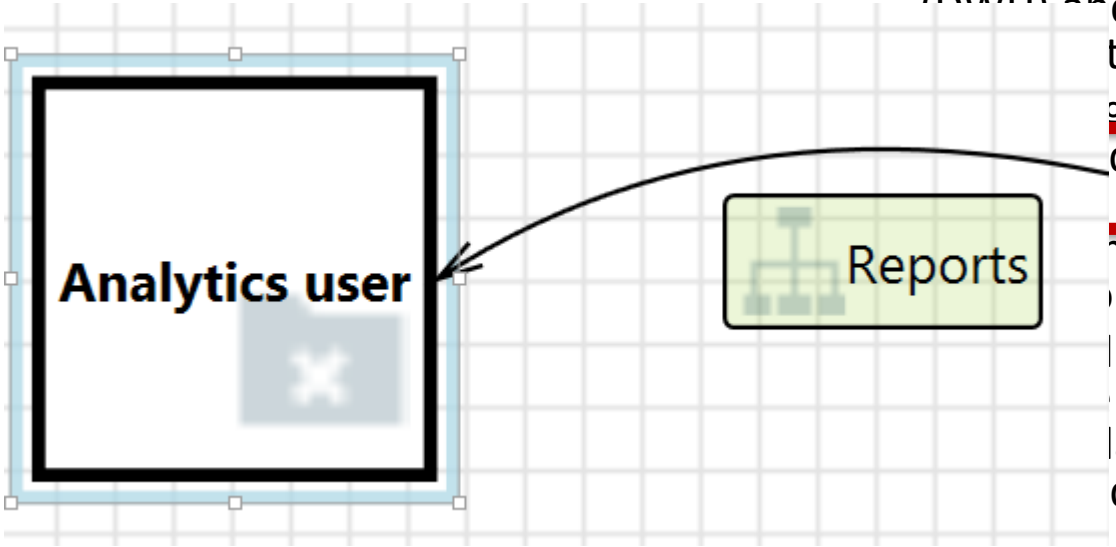


We are going to use a 3rd party reporting and analytics technology. They are going to host Data Warehouse (DWH) and reporting server on their infrastructure.

ve us licences to use their Analytics App, which can eporting server. The only ed to build in-house is an process,

et data from our database, and upload it to the DWH basis (they provide API for upload).

Last step

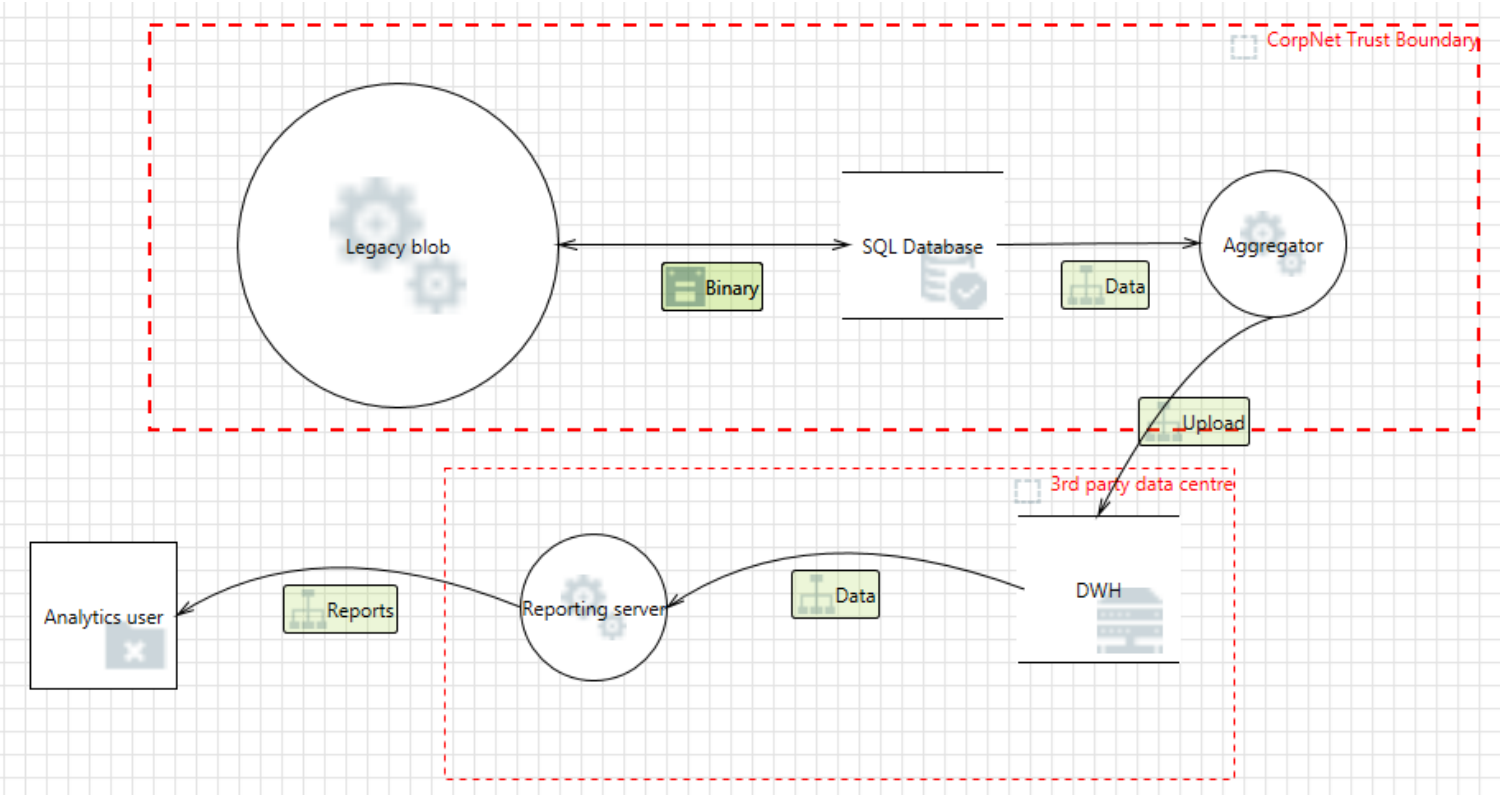


We are going to use a 3rd party reporting and analytics technology. They are going to host Data Warehouse (DWH) and reporting server on their ture.

give us licences to use their d Analytics App, which can reporting server. The only need to build in-house is an r process,

get data from our database, it and upload it to the DWH ar basis (they provide API for d upload).

Last step


CorpNet Trust Boundary
Legacy blob
Binary
SQL Database
Data
Aggregator
Upload
3rd party data centre
Analytics user
Reports
Reporting server
Data
DWH

Relevant Threats

Spoofing

- Can attacker upload data on our behalf? How we authenticate the destination before uploading?

Tampering and Information Disclosure

- Can attacker sniff the data or tamper with it?

Repudiation

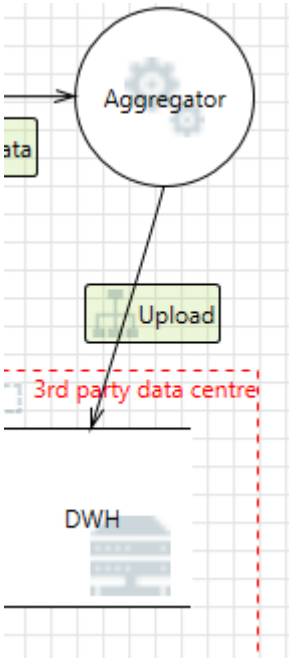
- Can DWH claim we didn't send the data? Or sent above the quota?

Denial of service

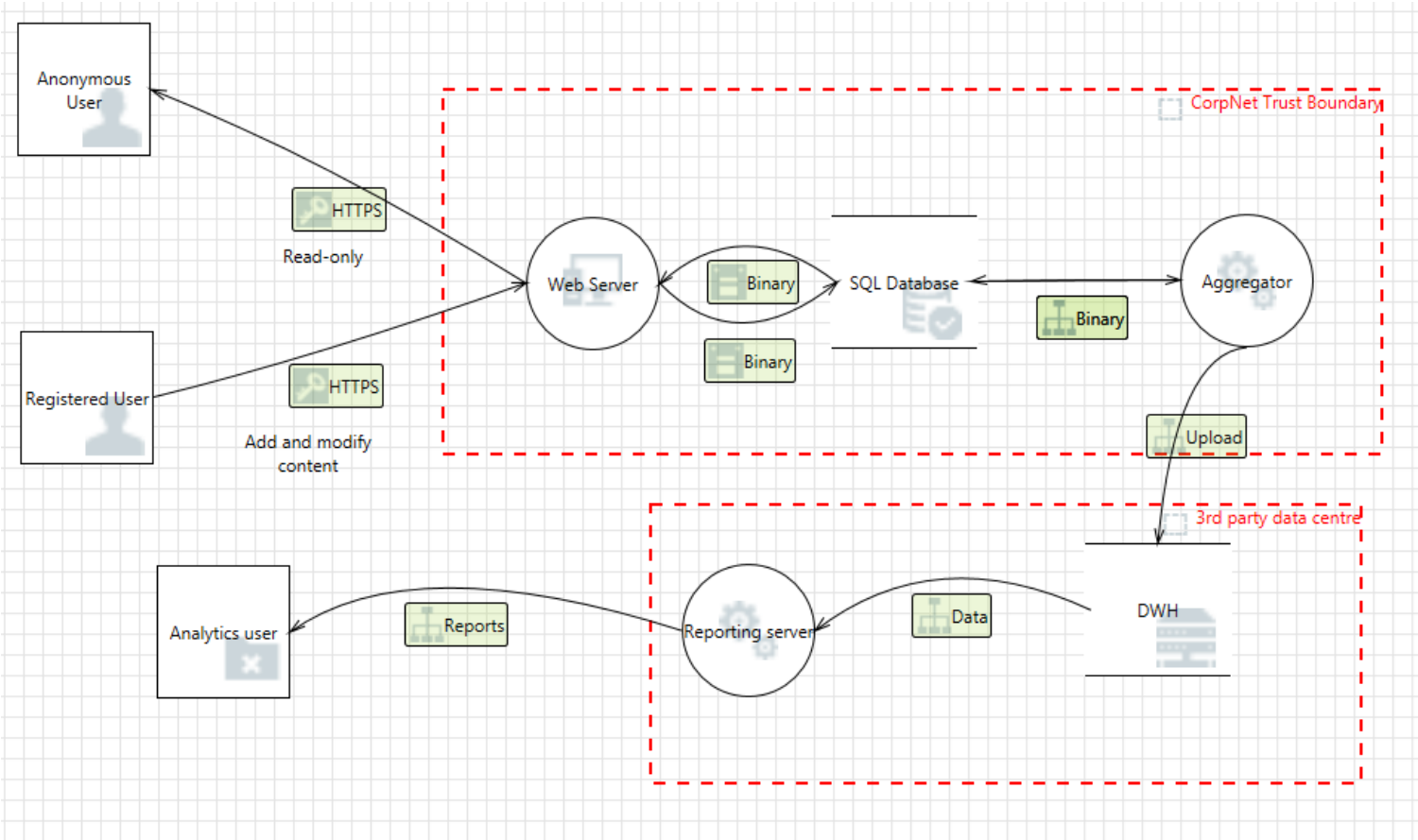
- Is there availability SLA for uploads?

Privacy

- Can our aggregation be reverse engineered?
- Do we need to notify the users that 3rd party is involved?



Irrelevant Threats



How to make them go away

- Can registered user inject malicious content?
  - We are not making it worse
- Can anonymous user bypass access controls and modify something?
  - We are not making it worse
- Is our datacentre infrastructure secure?
  - We are not making it worse (careful here!)
- Can analytics user abuse licencing?
  - Not our problem, 3rd party problem

Caveats

Not our problem

- If the team's task is not just to implement with a chosen provider, but to evaluate several providers.

We are not making it worse

- If you come across something so catastrophic in the "Legacy blob", that it's an immediately obvious critical flaw.

What if implementation deviates from design?

- Aggregator is implemented as two processes: one to read and aggregate the data, the other for actual upload

- Time pressure and we MUST have analytics in the release. Let's create a user for this 3rd party so they pull data directly from our DB.

Looks familiar?

1

Untested (ball of mud) legacy code

2

Introducing tests for new and modified code

3

Eventually getting (almost) fully tested code

This does not work in security!

- NTVDM bug – found in 2010, introduced in 1993
- Shellshock – found in 2014, introduced in 1989
- Heartbleed – found in 2014, introduced in 2011
- POODLE – found in 2014, existed since 1996
- JASBUG – found in 2015, introduced in 2000
- DROWN, Badlock, gotofail, etc.

Eventually need the whole picture

- What we don’t know can harm us
- The system is greater than the sum of its parts

Eventually is better than upfront

- People have developed the necessary skills
- Many subsystems will be already analysed
- Easier to achieve management buy-in

Conclusion

- Incremental threat modelling can fit any time-box, without disturbing the regular development cadence.
- You can build a model of the whole system in parallel, starting from day 1, or waiting for several cycles, whatever suits your situation.
- As a shortcut, you can bring external resources to help with the initial model.
- But for the best results in agile environment you have to involve the whole team.

Conclusion

Points of contact

Irene Michlin
Principal Security Consultant

M: +44 (0) 7972 333 148
E: irene.michlin@nccgroup.trust
T: @IreneMichlin

Thank You to Our Sponsors

CME Group

Hewlett Packard Enterprise

SYNOPSYS®

ARXAN

BLACKDUCK

CHECKMARX

CITRIX®

FLEXERA SOFTWARE

kiuwan

netsparker®

POSITIVE TECHNOLOGIES

RAPID7

Sonatype

Twistlock

Invest Northern Ireland

INTUITION TechChannels

jscrambler

PROMON

WhiteHat SECURITY

MCSGROUP
INTELLIGENT RECRUITMENT SOLUTIONS

edgescan

Invest Northern Ireland

Allstate
You're in good hands.