Thinking like an Attacker

Automated attack simulations and threat modeling with securiCAD

Erik Ringdahl

Product Director

foreseeti

© 2019, Foreseeti AB or its Affiliates. All rights reserved.

Who we are and why we are here

Foreseeti – Introduction

- Europe's leading provider of Automated Threat Modeling and Attack Simulation with securiCAD
- Founded in Stockholm, Sweden in 2014. Background research started 2000 at KTH Royal Institute of Technology
- Customers in Europe, US, Singapore and Australia within defense, critical infrastructure, finance

SOCCRATES

- Generate models of ICT Infrastructures to provide and analyze attack graphs based on the models
- Suggest mitigation strategies based on attack graphs

Agenda

- Capital One Breach: What happened and why
- Thinking in Graphs
- Simulating attacks on AWS environments: Graph-thinking in practice

© 2019, Foreseeti AB or its Affiliates. All rights reserved.

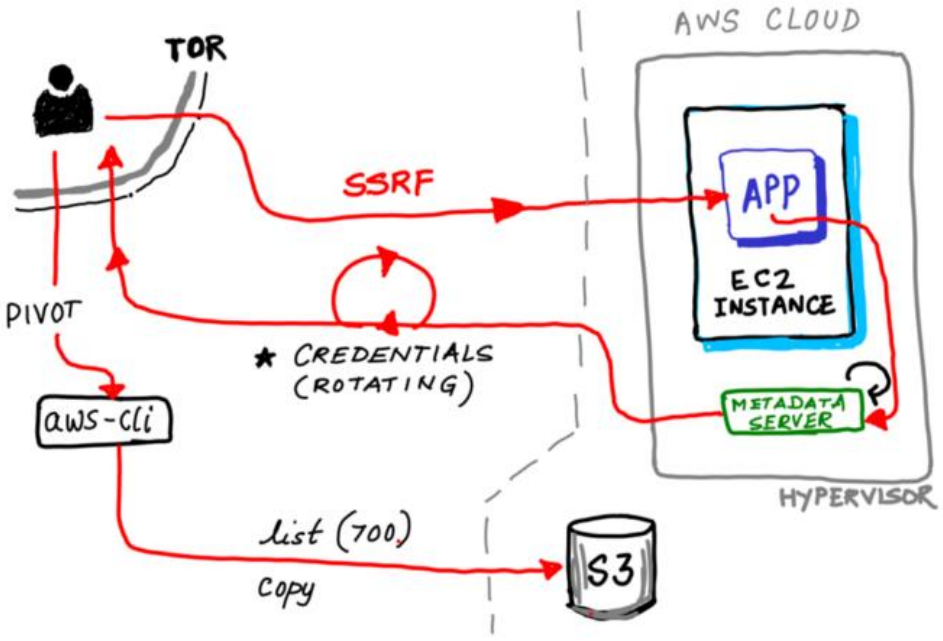
A recap of the Capital One breach

What happened?

Capital One has acknowledged an attacker breached 140,000 Social Security numbers and 80,000 bank account numbers

- Gaining access to an EC2 instance with misconfigured firewall
- Query EC2 instance metadata service to acquire IAM role name
- Using role name to extract credentials
- Using this IAM role policy, the attacker accessed S3 storage service
- Used S3 commands to copy data from the roles accessible bucket

How could it happen?



https://blog.shiftleft.io/capital-one-breach-crime-board-case-of-speculative-sleuthing-e18fa937fa21

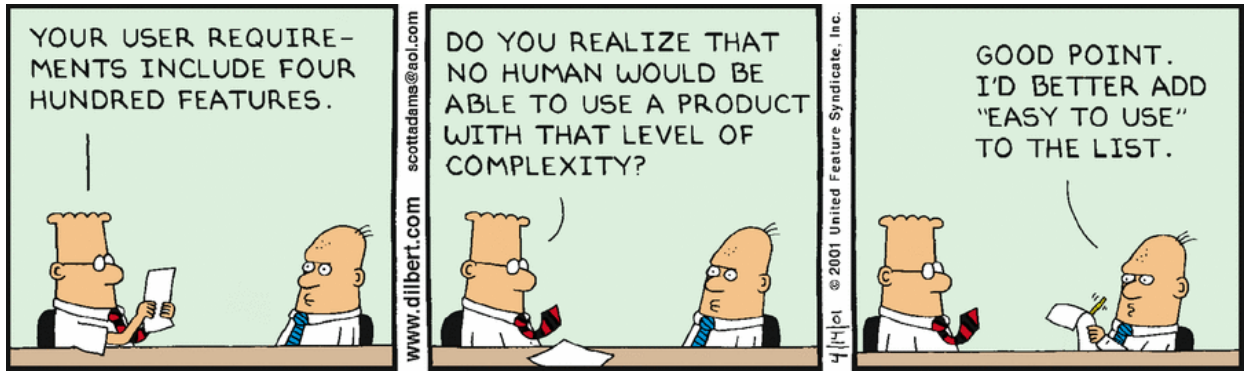
© 2019, Foreseeti AB or its Affiliates. All rights reserved.

Too many knobs and switches

Risky Business RB550

”Capital One are known for being really smart when it comes to this stuff, so when you see them getting owned by an attacker who falls into the category of Internet jerk, it does give you a moment.”

”Yes, part of the problem is that the Amazon product suite really is super complicated. You can build all sorts of amazing things at really big scale but doing it right consistently is really, really hard.”



https://dilbert.com/strip/2001-04-14

© 2019, Foreseeti AB or its Affiliates. All rights reserved.

Attackers think in Graphs, Defenders think in Lists

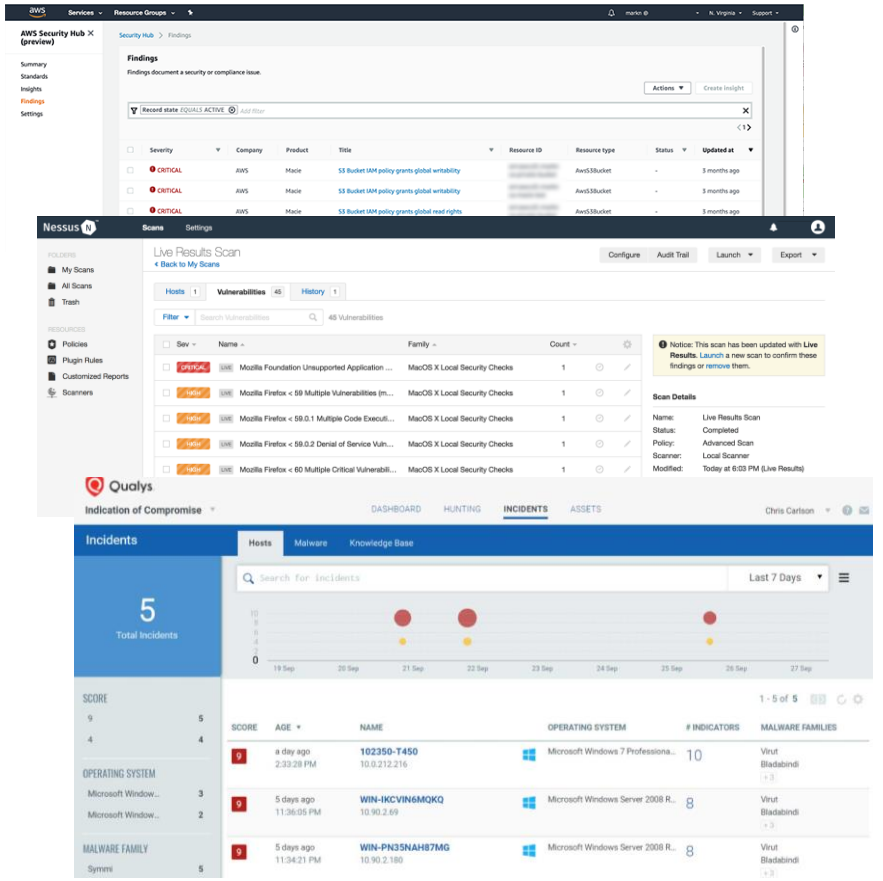
The difference between an attacker and defender mindset

In an article from 2015, John Lambert engineer at Microsoft's threat intelligence center, wrote about the difference between an attacker and defender mindset:

"A lot of network defense goes wrong before any contact with an adversary, starting with how defenders conceive of the battlefield"

"Most defenders focus on protecting their assets, prioritizing them, and sorting them […] Defenders are awash in lists of assets—in management services, inventory databases [...] There's one problem with all of this"

"Defenders don't have a list of assets—they have a graph"



© 2019, Foreseeti AB or its Affiliates. All rights reserved.

Defenders need to think in Graphs too!

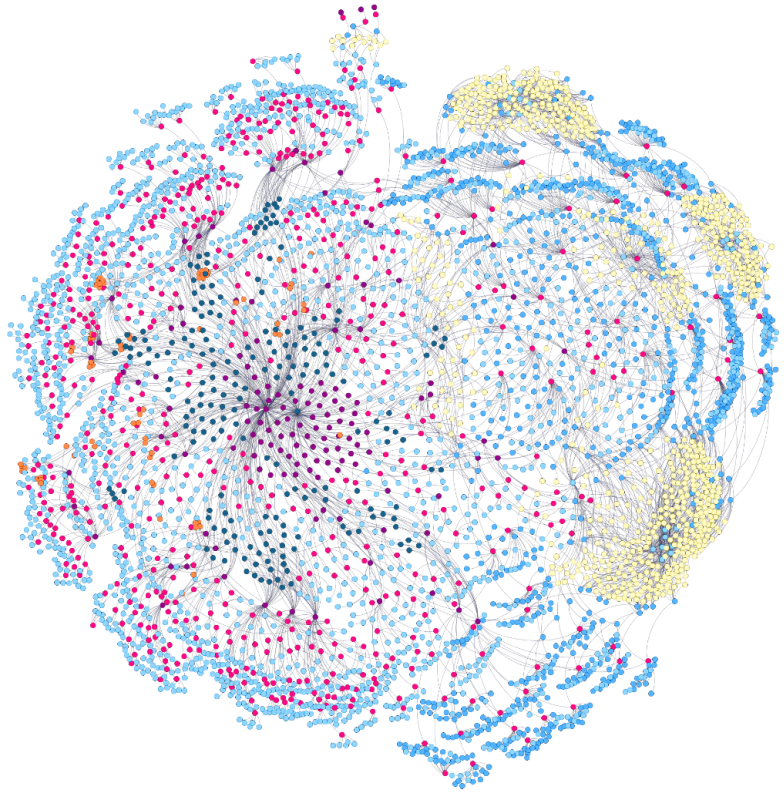
Adopting the Attacker mindset

Defenders can have the upper hand when visualizing the battlefield with full information about their own environment, wheares attackers need to study the network piece by piece. However:

“Defenders should take a lesson from how attackers come to understand the graph. Attackers study the infrastructure as it is—not as an inaccurate mental model…” – John Lambert

Leveraging available data

With AWS environments, we have the luxury of APIs that describes our whole existing environment, its configuration, accounts, access rights etc. We can leverage that data to generate a graph of our environment and all of its dependencies and influences



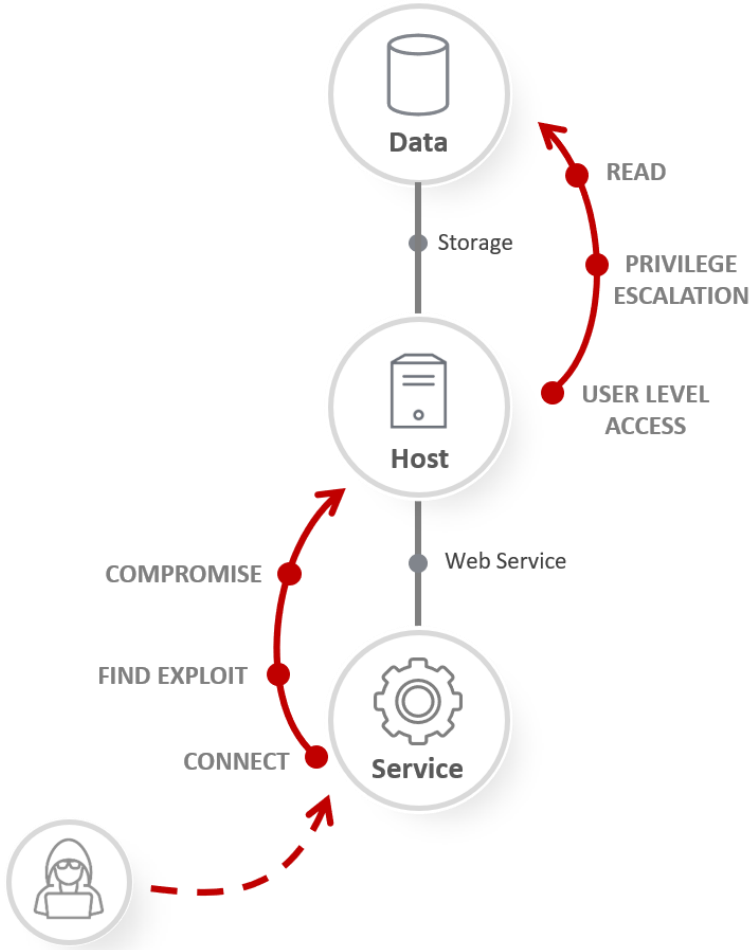
© 2019, Foreseeti AB or its Affiliates. All rights reserved.

Gaining the Attacker view of the Battlefield

Threat Modeling

But our graph is not complete yet. We still need to figure out how to see the graph from the attacker's viewpoint i.e., what can be exploited and how can an attacker pivot in the graph?

The concept of Threat Modeling is about thinking like an attacker – to figure out what combination of attack steps an attacker might exploit – in order to mitigate them in the most effective way, before an actual attacker exploits them



Problems with Threat Modeling

- Manual and labor intensive
- Complex process
- Biased assessments

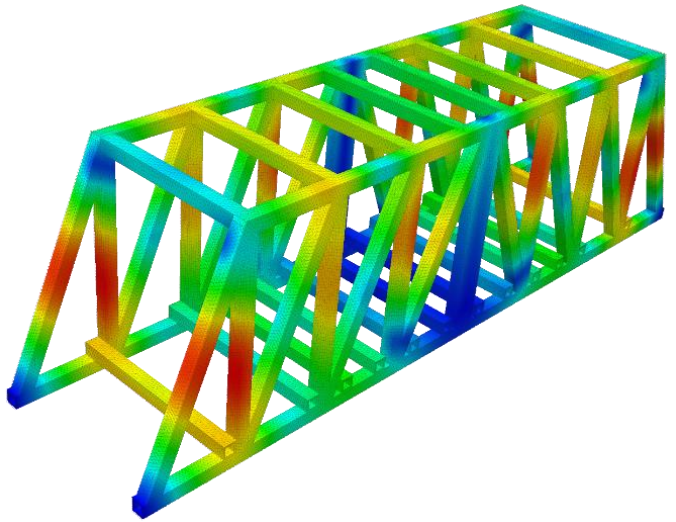
Overwhelming for large dynamic environments

© 2019, Foreseeti AB or its Affiliates. All rights reserved.

Thinking like an Attacker, the easy way

Thinking is exhausting, lets simulate!

Simulations and computer aided design are a key part of most other engineering domains. Let's say that you are designing a new bridge, you then create a model of that bridge in a CAD tool and then simulate mechanical stress on the model to see if there are any weaknesses in the design and how all parts of the bridge influence each other



Simulating attacks on AWS

By defining a set of "laws" for AWS i.e., what attacks steps an attacker will be able to take inside and between assets in our environment, we can simulate attacks on AWS environments, just like simulating an earthquake on a model of a bridge

Most attack steps will, in fact, be legitimate actions in AWS. For example, an attacker may exploit the privileges of an IAM role assigned to a compromised EC2 Instance

© 2019, Foreseeti AB or its Affiliates. All rights reserved.

The path of Least Resistance

We’re all lazy

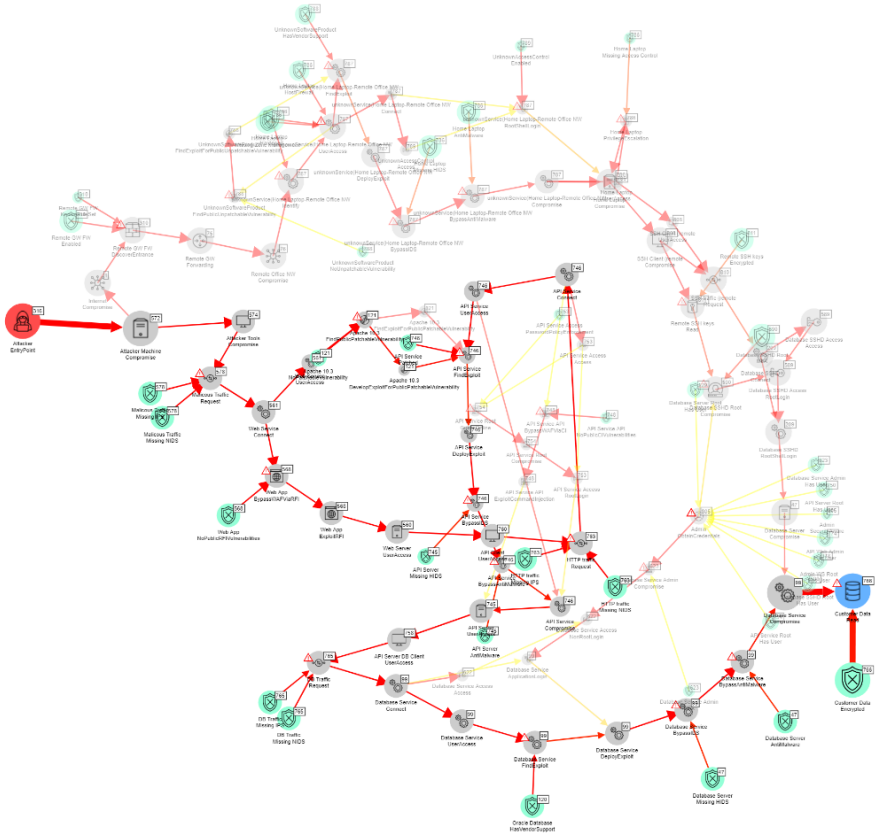
Of course, this isn’t unique to humans. Everything takes the path of least resistance: water, electricity, and hackers – they'd much rather look for the open windows and doors in their targets from sloppy passwords or misconfigurations

By using Monte Carlo simulations we can simulate thousands of attacks on the environment to find the paths of least resistance i.e., the most likely paths, from the attacker´s entry point, through our AWS environment, to our high value assets

Entering The Matrix

We have now opened our minds to the power to think like an attacker and see the world as it really is, full of graphs, ready to exploit

How do we put it into practice?



© 2019, Foreseeti AB or its Affiliates. All rights reserved.

A formal approach to laziness

MAL (Meta Attack Language)

The Meta Attack Language is, as its name suggests, a meta language that specifies all the rules and the elements for domain specific languages. It is an efficient way to avoid creating new attack graphs for every different case on the domain under study and is a open source project supported by KTH and foreseeti

https://mal-lang.org

```
grammar Mal;

compilationUnit: (associations | category | include | langMeta)* EOF;

// Revisit where meta is placed?
meta: metaType Colon String;
metaType: Info | Rationale | Assumptions;
type: LeftBracket Identifier RightBracket;

include: Include (Identifier | Dot | Separator)+;

langMeta: Defense Identifier Colon String;

associations: Associations LeftBrace association* RightBrace;
association: Identifier type multiplicity LeftRelation Identifier RightRelation multiplicity type Identifier meta*;
multiplicity: Single | ZeroSingle | SingleMany | Many;

category: Category Identifier meta* LeftBrace asset* RightBrace;

asset: assetType Identifier (Extends Identifier)? meta* LeftBrace attackstep* RightBrace;
assetType: Asset | AbstractAsset;

attackstep: attackstepType Identifier ttc? meta* existence? (reachedType statement (Comma statement)*)?;
attackstepType: AttackstepAll | AttackstepAny | Defense | DefenseExistence | DefenseNonExistence;
```

DSL (Domain Specific Language)

By using the MAL framework we can create languages for specific domains with formal definitions of the domains assets, associations, attack steps and defenses

For today’s presentation we will focus on a DSL for AWS

```
55    asset Statement
56    info: "A policy is an entity in AWS that, when associated with an ide
57    {
58      | satisfy
59        -> performAction
60
61      & performAction
62        -> actions.perform
63
64      # deny
65        info: "If this defense is activated, the action will be prohibite
66        rationale: "When competing policies conflict, deny wins over allo
67        -> performAction
68    }
```

© 2019, Foreseeti AB or its Affiliates. All rights reserved.

There's an app for that!

Introducing securiCAD

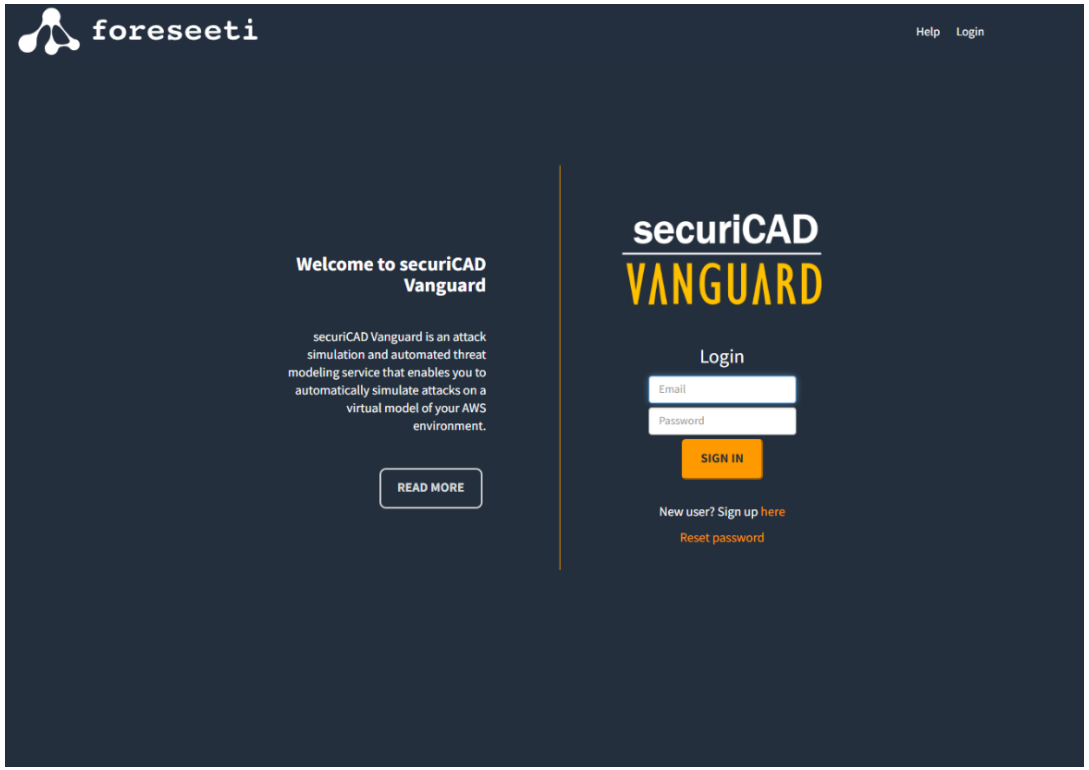
Generate model

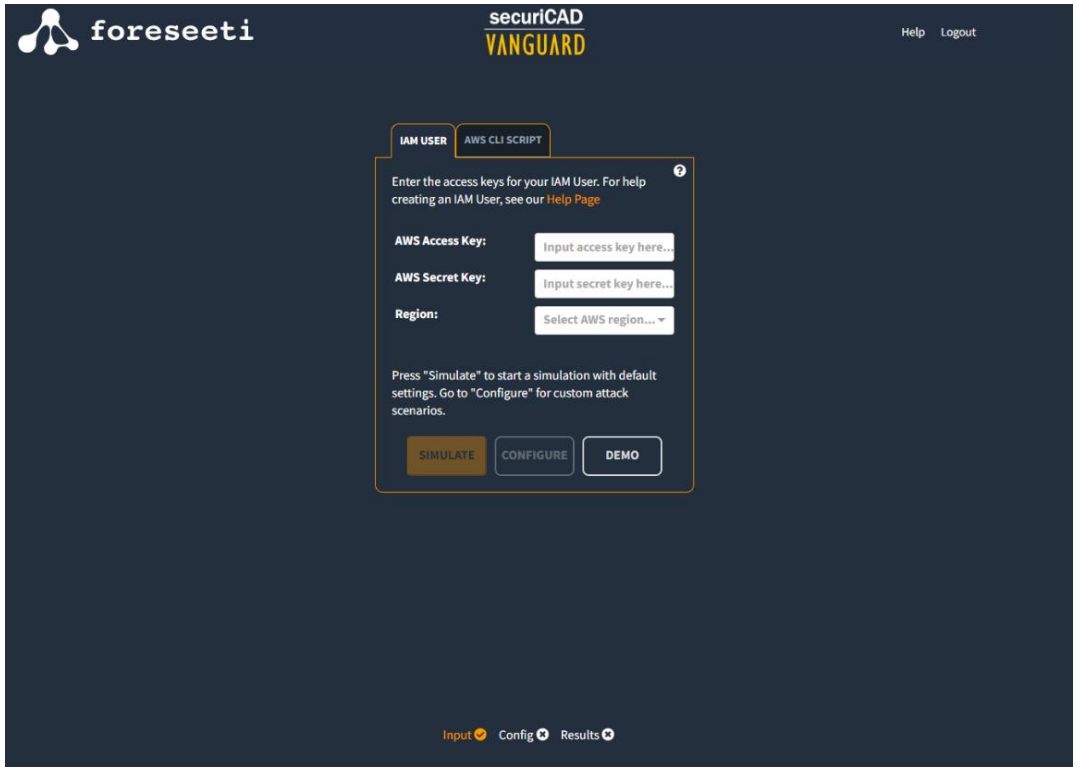
Simulate attacks

Manage risk exposure

© 2019, Foreseeti AB or its Affiliates. All rights reserved.

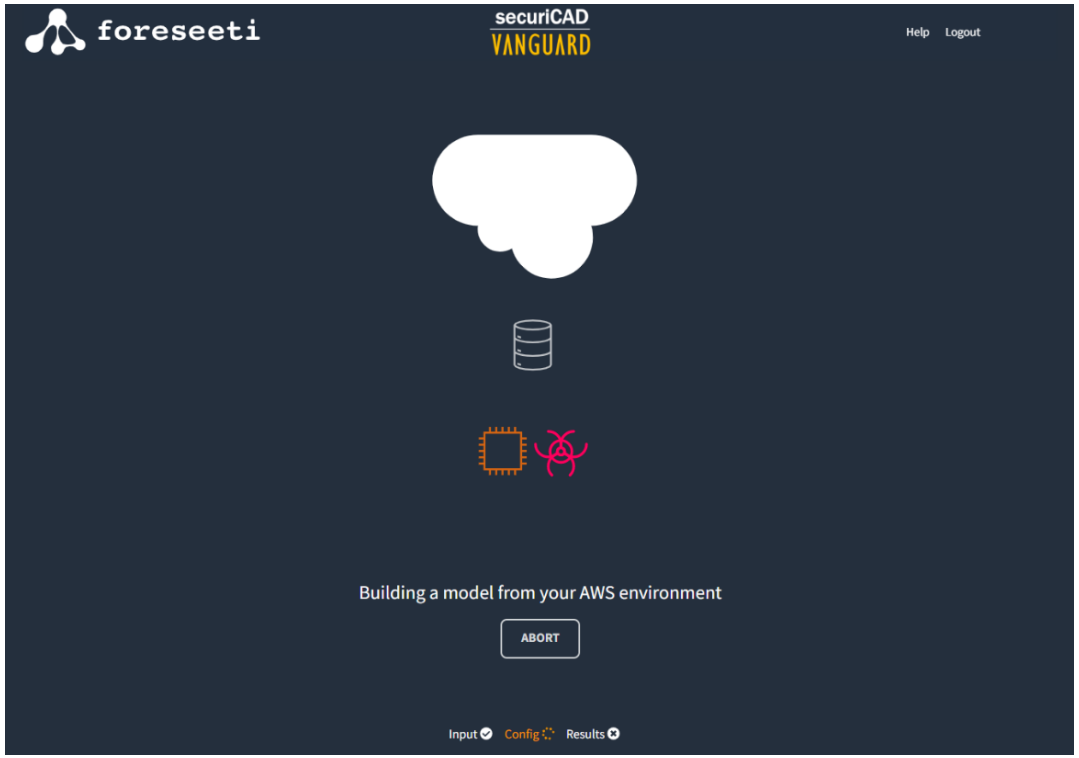
Simulating the Capital One breach

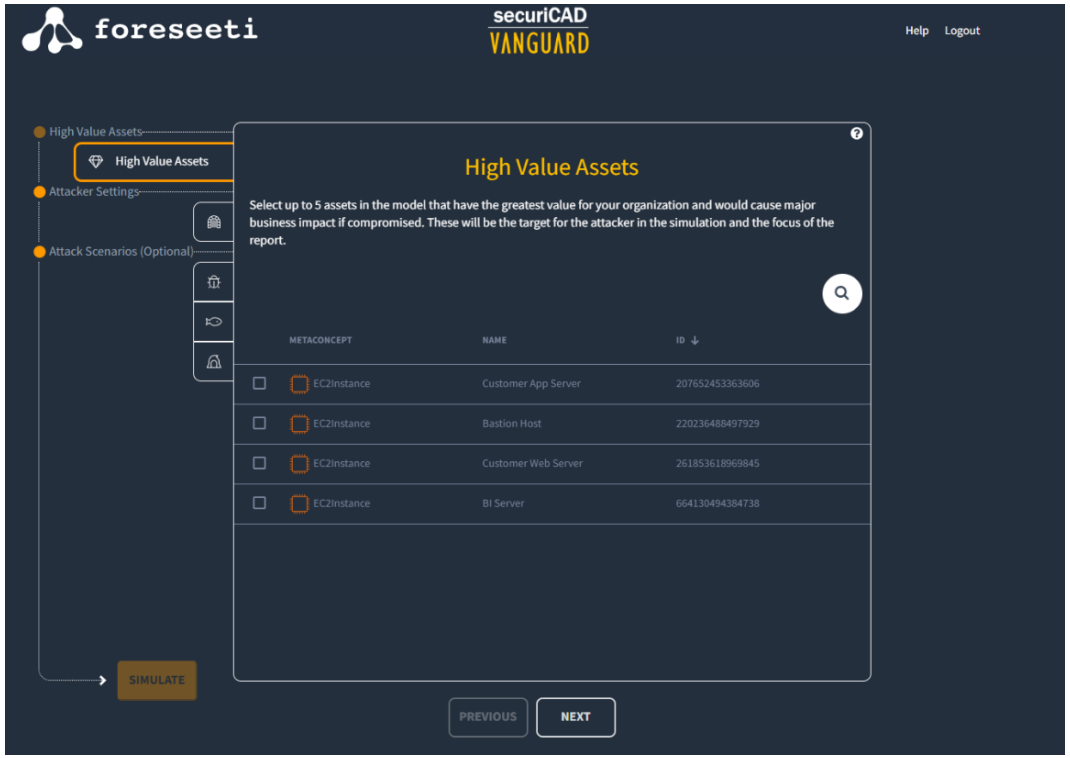




© 2019, Foreseeti AB or its Affiliates. All rights reserved.

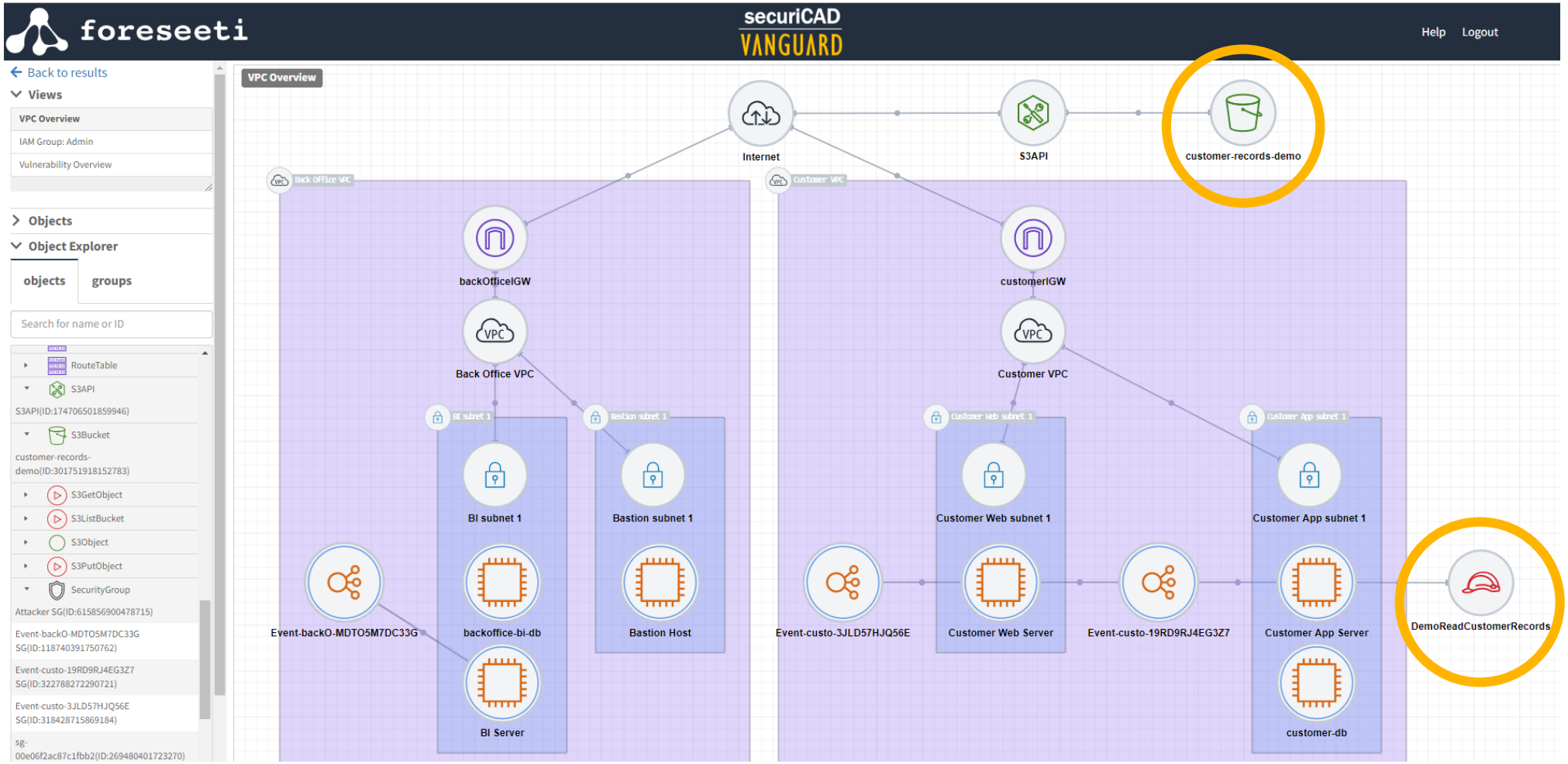
Simulating the Capital One breach





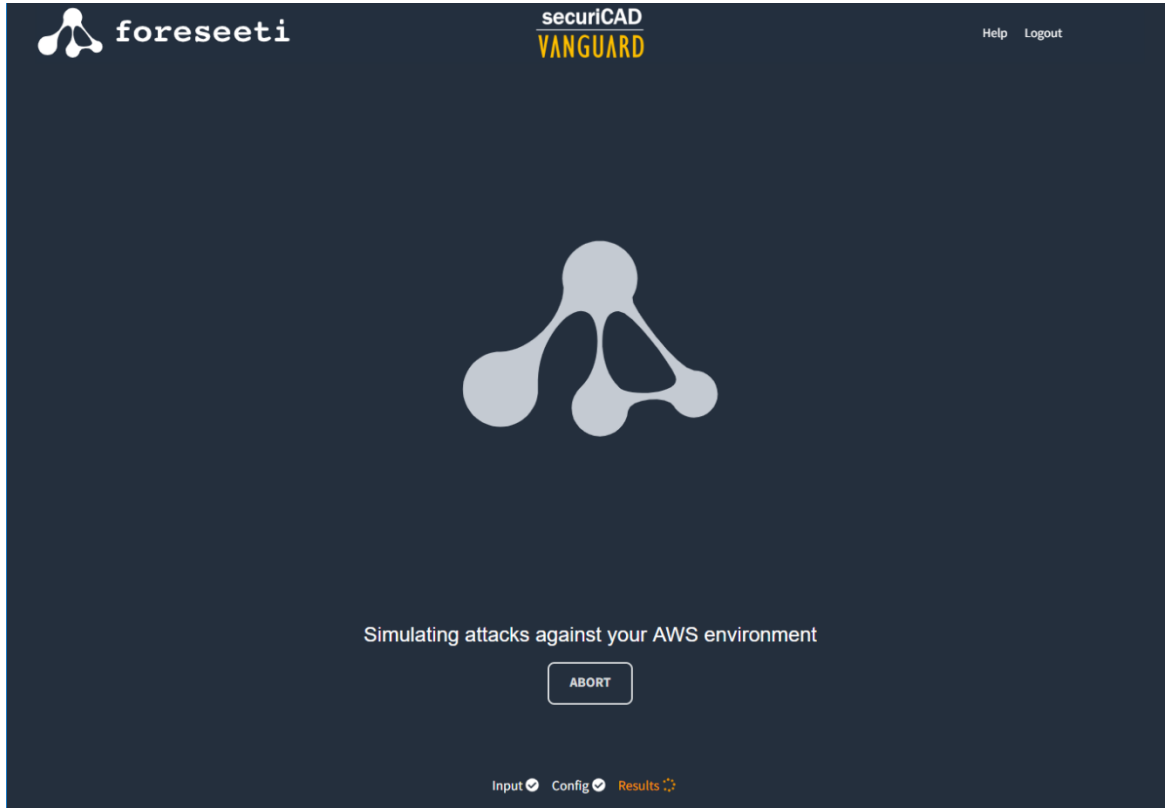
© 2019, Foreseeti AB or its Affiliates. All rights reserved.

Simulating the Capital One breach



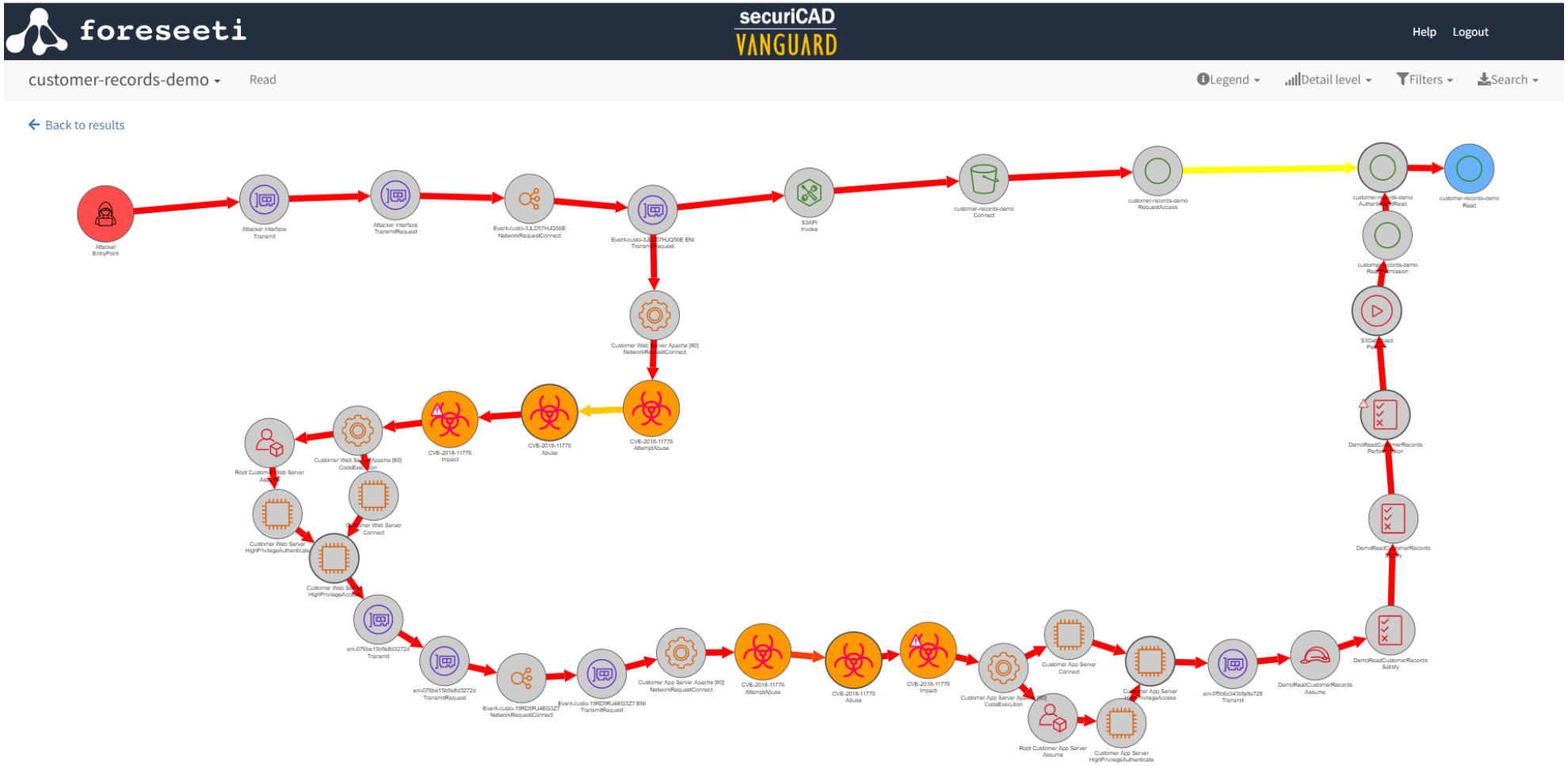
© 2019, Foreseeti AB or its Affiliates. All rights reserved.

Simulating the Capital One breach



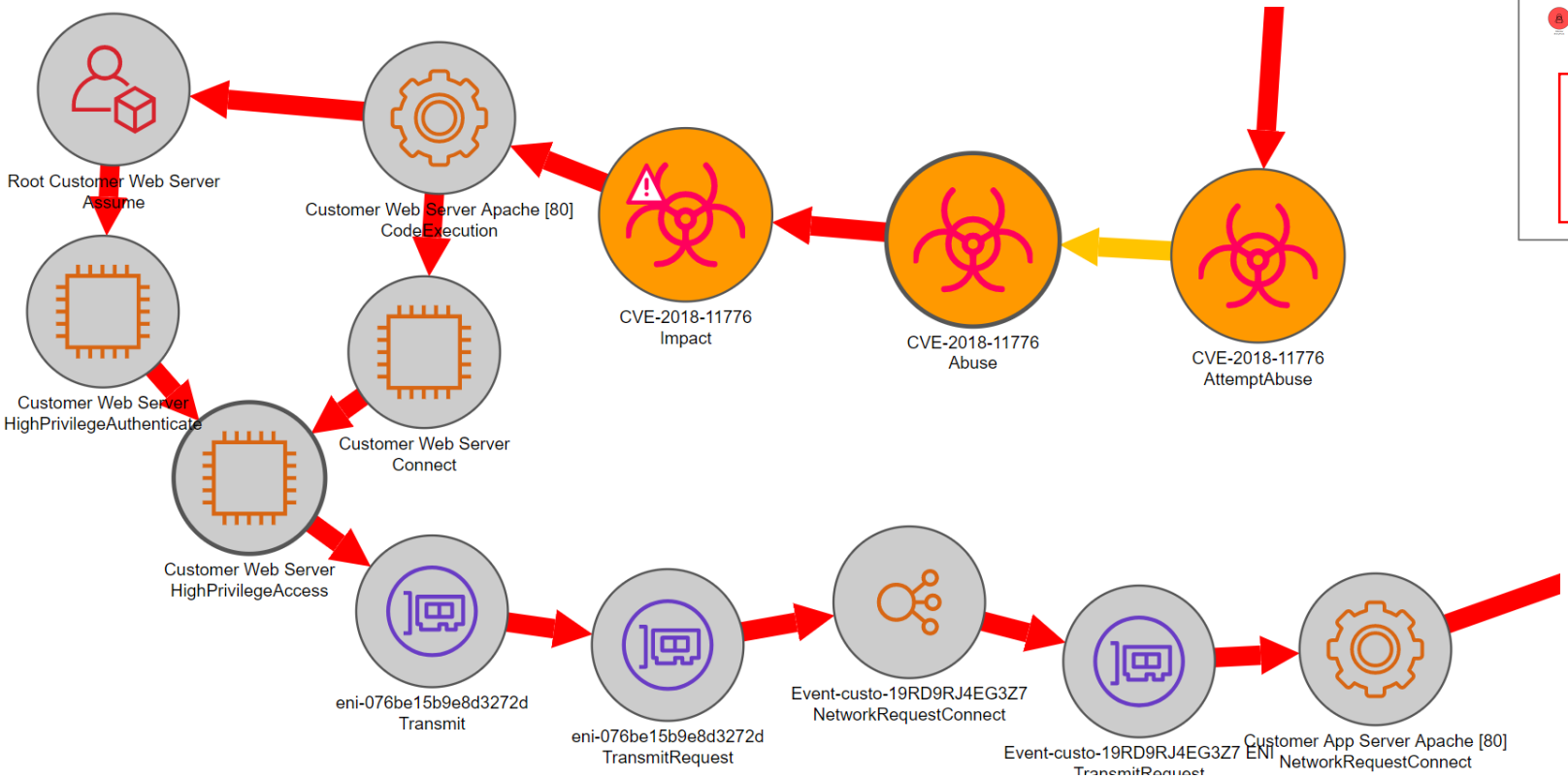
© 2019, Foreseeti AB or its Affiliates. All rights reserved.

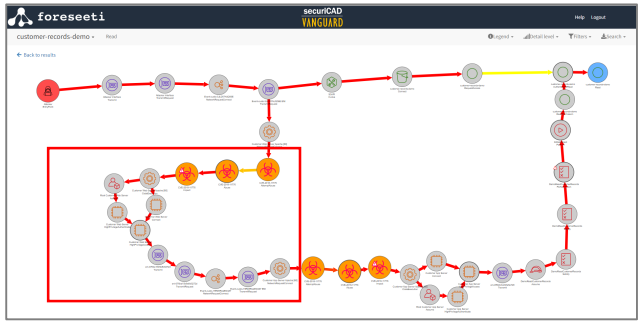
Simulating the Capital One breach



© 2019, Foreseeti AB or its Affiliates. All rights reserved.

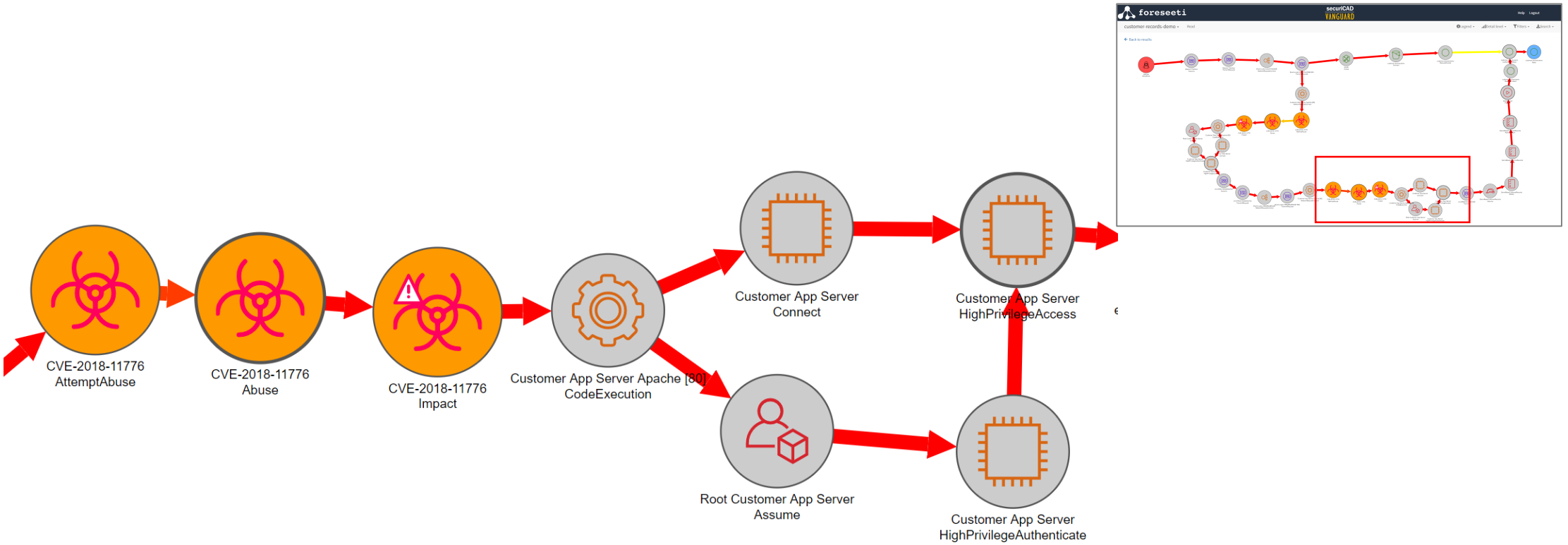
Simulating the Capital One breach





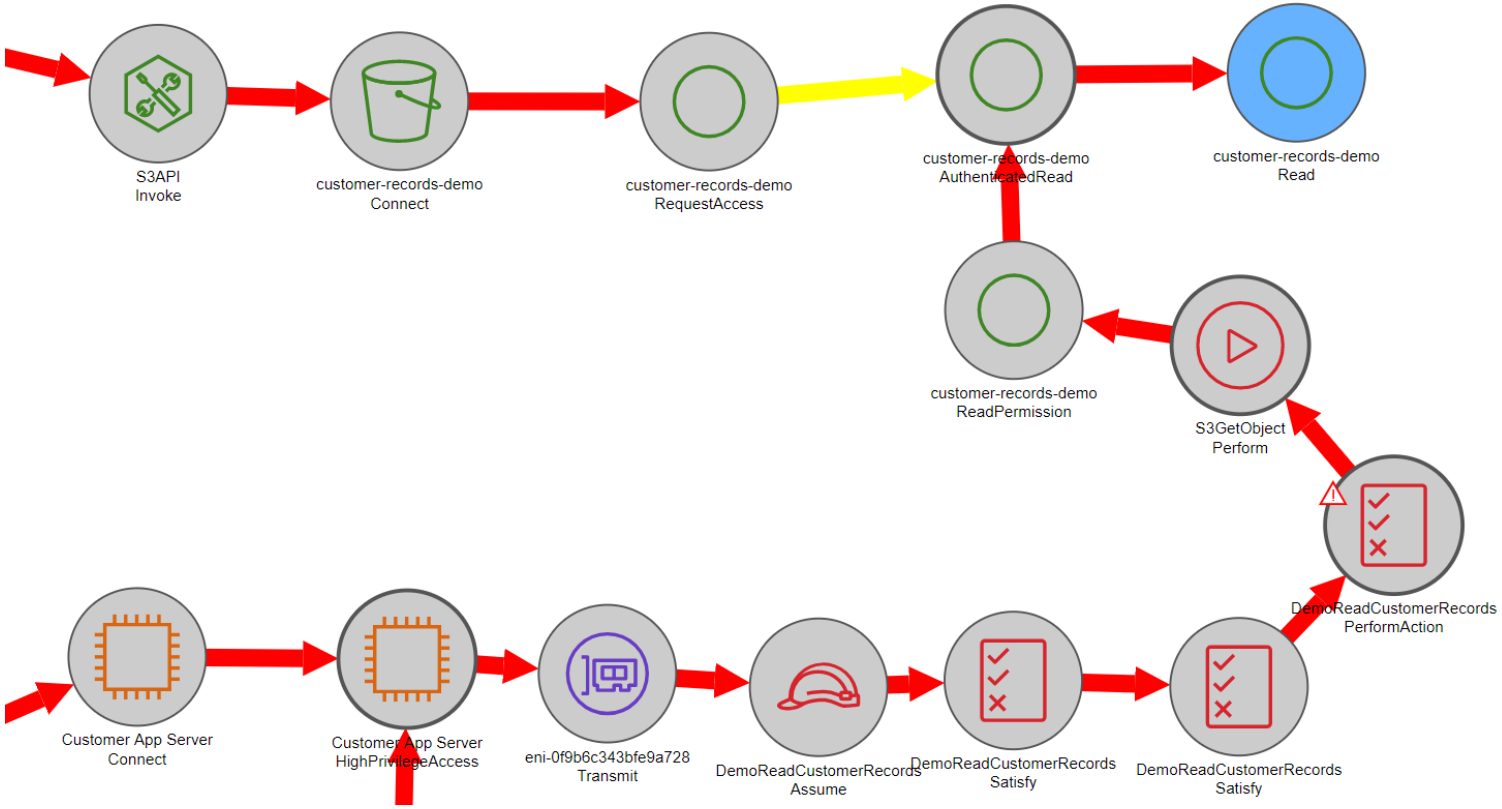
© 2019, Foreseeti AB or its Affiliates. All rights reserved.

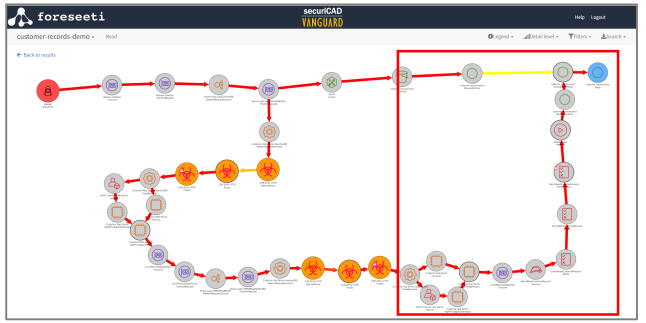
Simulating the Capital One breach



© 2019, Foreseeti AB or its Affiliates. All rights reserved.

Simulating the Capital One breach





© 2019, Foreseeti AB or its Affiliates. All rights reserved.

Automated threat modeling and attack simulations

In summary

With securiCAD and MAL we want to help defenders understand the consequences of their actions and weaknesses in their IT environments

By leveraging known processes such as threat modeling and Monte Carlo simulations and automating them

To aid in prioritizing mitigations and giving defenders a fighting chance at winning the war against the internet jerks

Free to try now!

- securiCAD Community Edition
- securiCAD Vanguard Early Access

https://www.foreseeti.com/community/

© 2019, Foreseeti AB or its Affiliates. All rights reserved.

Thank you!

Erik Ringdahl

erik.ringdahl@foreseeti.com

© 2019, Foreseeti AB or its Affiliates. All rights reserved.