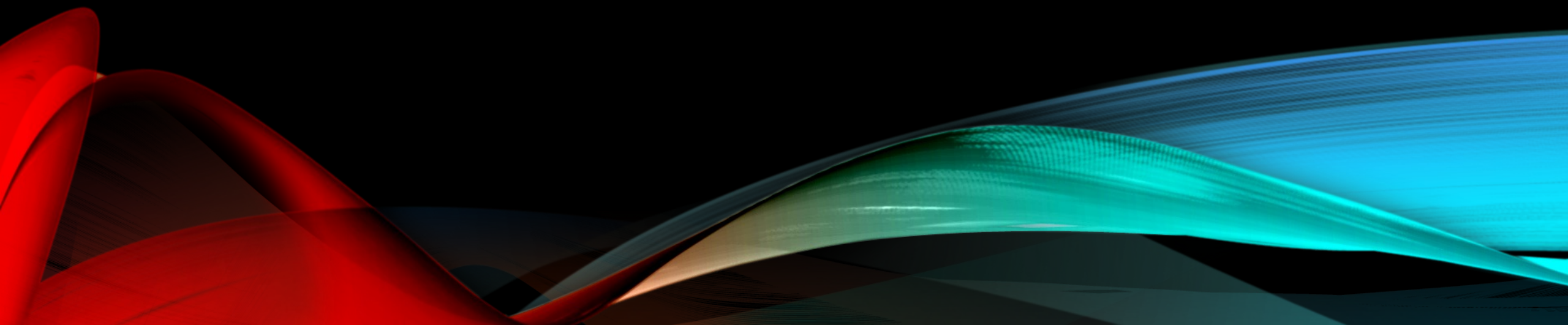
Proof of Concept

BUILDING A HYPERSECURE DEVELOPMENT ENVIRONMENT

Rene Malmgren RedToken 2025-09-26

WELCOME TO RENES TALES OF CYBER HORROR.



WHO IS THIS?

SECURITY RESEARCHER, FORENSIC ANALYSIST, AND BLOCKCHAIN ENTHUSIAST.

Rene Malmgren
Ericsson Security Program 2020-2024

DID MY STORY SCARE YOU?

EXAMPLE OF AN ATTACK

- Type: Target Access Operations
- Codename: Operation socialist
- Attacker: National state actor
- Target: National Carrier
- Objective: Billing information, staging for penetration of customers communication.
- Duration: 2010-2013
- Multi layered attack.
  - First stage of the attack starts with redirecting the victim to a compromised webpage page to gain access to the computer.
  - Second stage deploys a rootkit on the device to gain full administrative access.

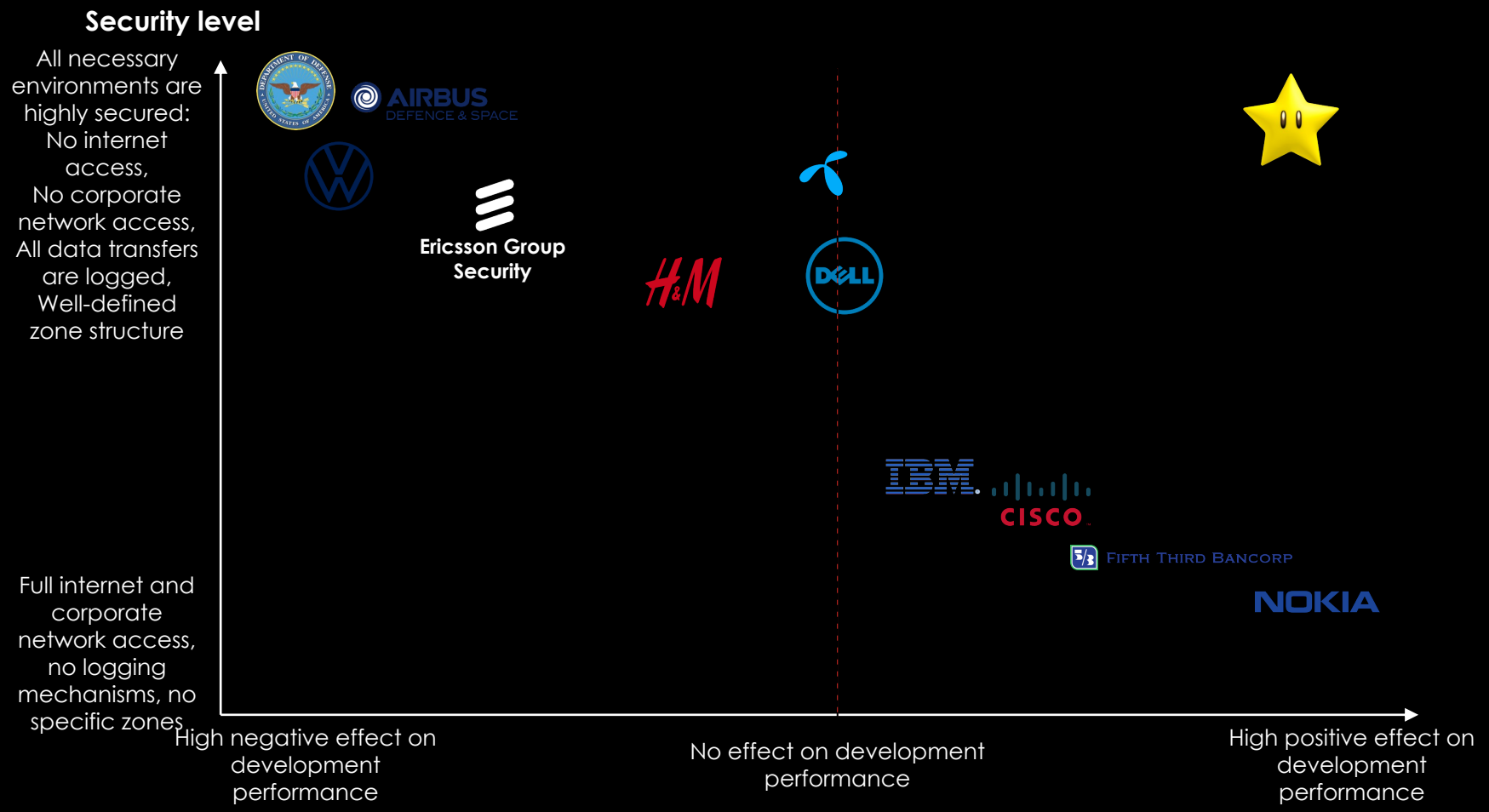
WHAT WORKS, WHAT DOES NOT.

- Password based authentication
  - Do I really have to comment?
- Any OS level administrative access control.
  - They have root kits as COTS.
- Cloud bases authentication services.
  - You cant outsource your problems, only insource others.
- Fine grained access control.
  - No way you can understand what you are doing.
- Any closed source stuff.
  - They have been caught so many times to sell vulnerabilities.
- Complex stuff.
  - If you cant audit it its not safe.
- VDI.
- Free beer stuff.
  - If you are not paying for it, you are not the customer, you are the product for sale.
- Pain.
  - If its not convenient its not secure.
- Access tokens
- Hard information segregation.
  - All the military work with this.
- Public, blockchains.
  - Billions at stake, work well.
- Correctly configured crypto.
  - We have seen them work very hard to disable them.
- Hierarchical Keystore.
  - All blockchain wallets use them.
- Type 1 hypervisor.
  - Requirement from NCSC (Uk).
- Defence in debt.
  - Every military does this.
- Correctly configured 2FA.
  - Surprisingly.
- Rust.
- Guardian protocols:
  - TLS, SSH, GPG, IPSEC.

HARD INFORMATION SEGREGATION

- The segregation **must** withstand a full exploit (root level) of the office environment without posing any risk to the information in the development environment.
- The driving force behind segregation is to provide an adequate protection for the **integrity**, and to a lesser extent the **availability** of the software.
- A logical segregation is seen as adequate.

MARKET COMPARISON



CHALLENGE FROM DEVELOPERS

- Must be able to work efficiently
  - Must support reasonable performance
  - Must support reasonable functionality
    - Shared clipboard
    - File transfer capability
    - Screen sharing capability
  - Must support up to date and productive software
  - Must support a broad ability for automatic self-service

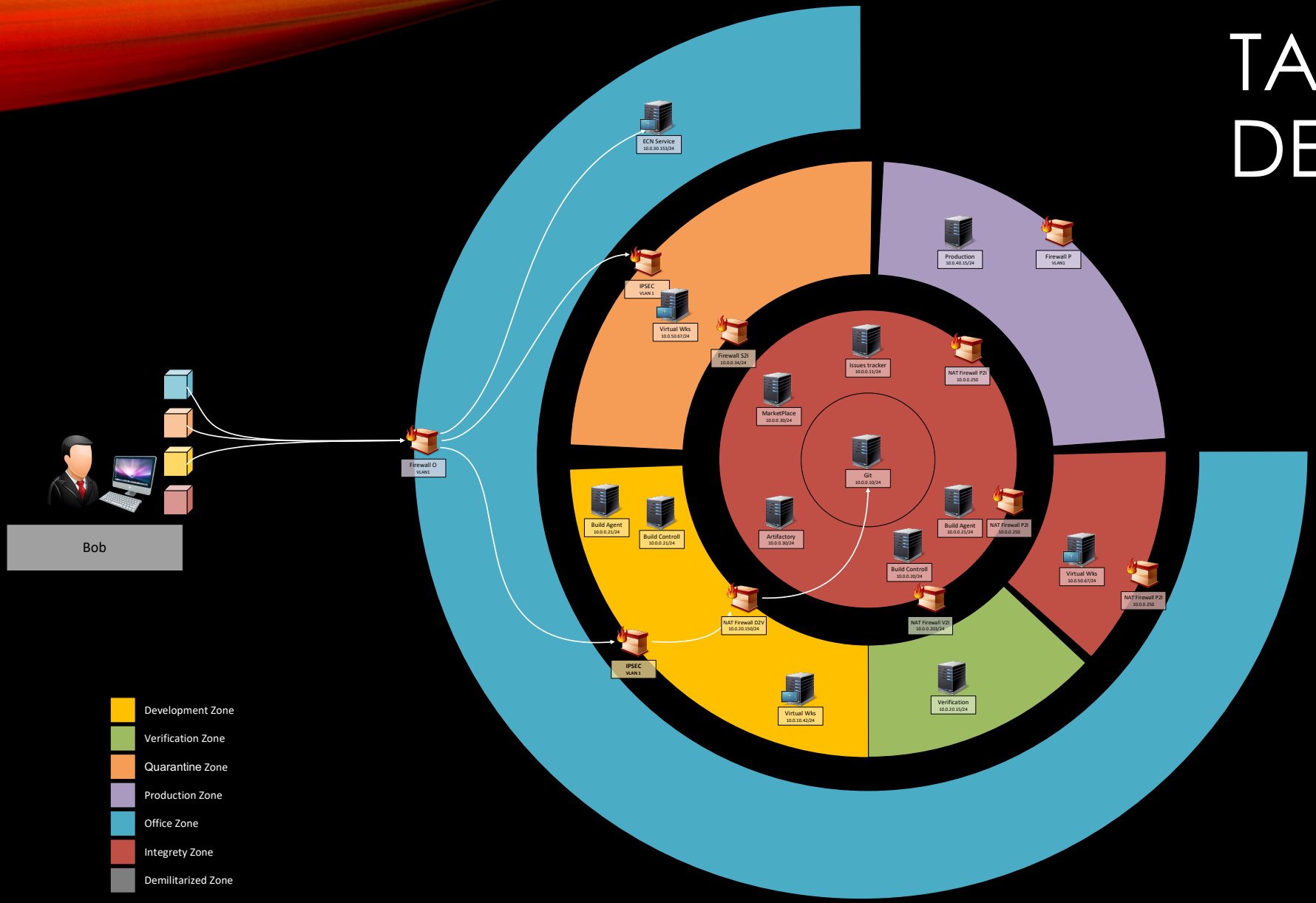
GUARDIAN PROTOCOLS

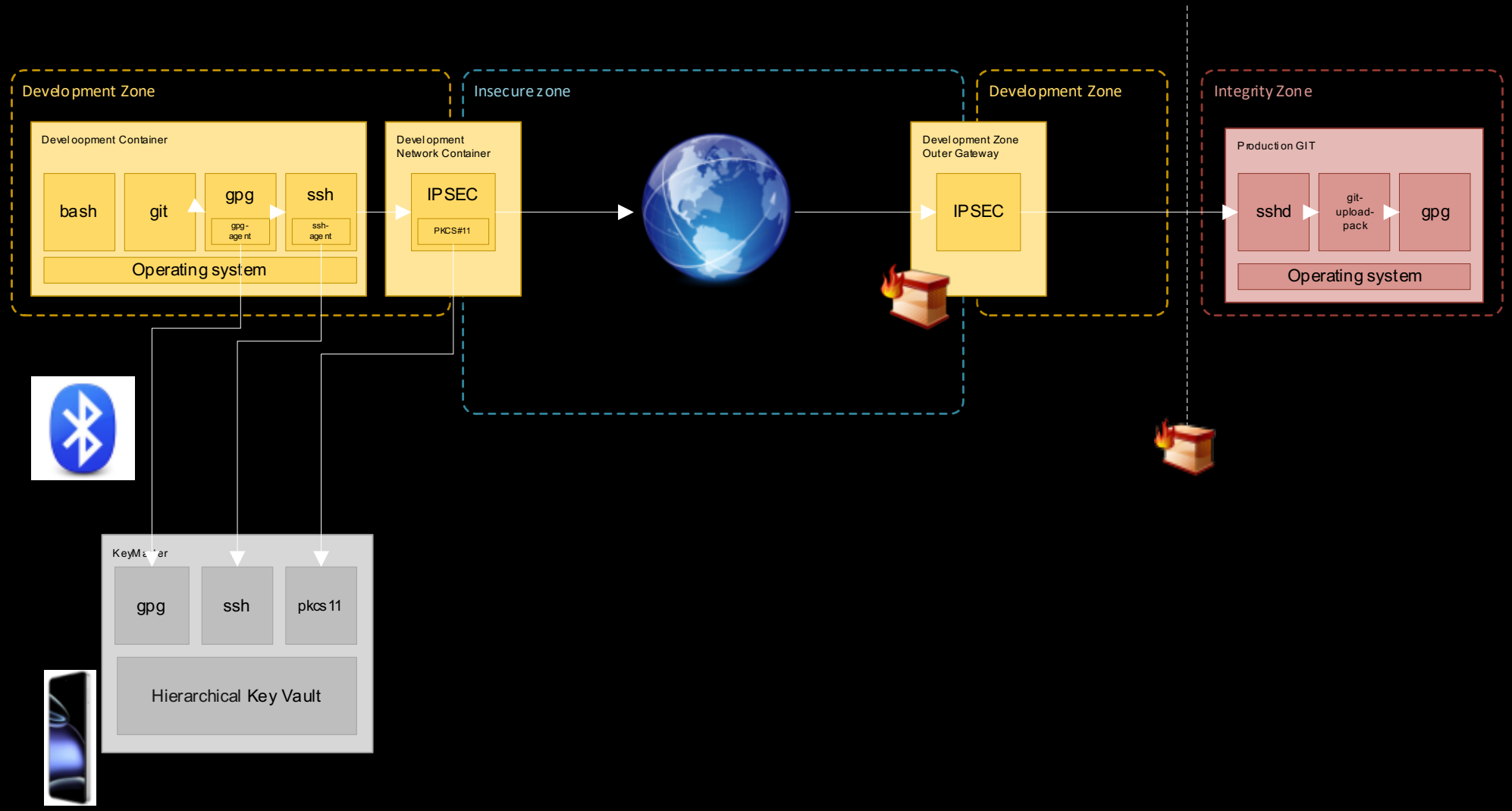
- IPSEC
- GPG
- SSH
- TLS

WHAT WE WOULD LIKE

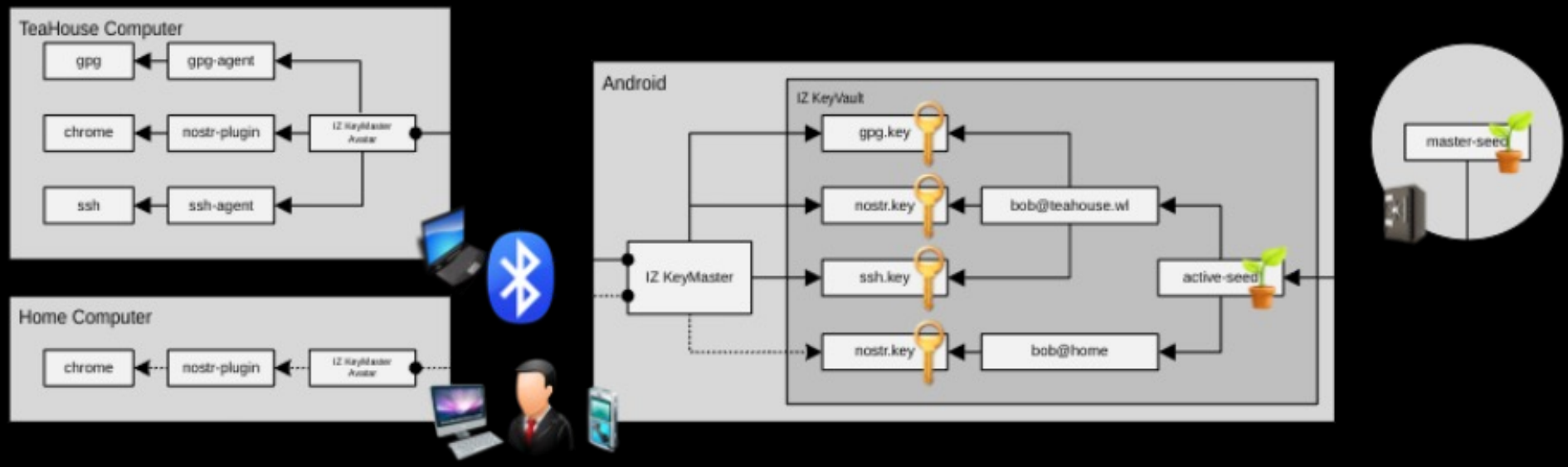
- Withstand 2 00-vulnerabilities in guardian protocols and still keep software integrity intact.
- Be able to restore the integrity of the software from a full exploit.
- Work reasonably efficiently.

TARGET DESIGN




Development Zone
Develoopment Container
bash
git
gpg
gpg-agent
ssh
ssh-agent
Operating system
Development Network Container
IPSEC
PKCS#11
Insecure zone
Development Zone Outer Gateway
IPSEC
Development Zone
Integrity Zone
Production GIT
sshd
git-upload-pack
gpg
Operating system
gpg
ssh
pkcs11
Hierarchical Key Vault

IZ KEYMASTER



CONTROLLED INFORMATION TRANSFER EXAMPLES

Information **may** be transferred by the developer from the **office zone** to the **development zone** using the clipboard.

Information **may** be transferred by the developer from the **development zone** to the **office zone** using the clipboard.

Information **may** be transferred by the developer from the **office zone** to the **development zone** if it's contained in a single file with known content.

Information **may** be transferr ed by the developer from the **development zone** to the **office zone** if it's contained in a single file with known content.

Information in the **development zone may** be showed by the developer to a college using screen sharing software like teams running in the **office zone**.

Information **must not** be transferred by the developer to the **development zone** if it contains **complex**, or **unknown** information. This information must be packaged and checked in using a **quarantine zone**.

Information **must not** be transferred by the developer from the **development zone** if it contains **complex**, or **unknown** information, without prior authorization.

Information **must not** be transferred from, or altered in the **development zone** by a scrip running in the **office zone** in the background when the developer is unaware of it.

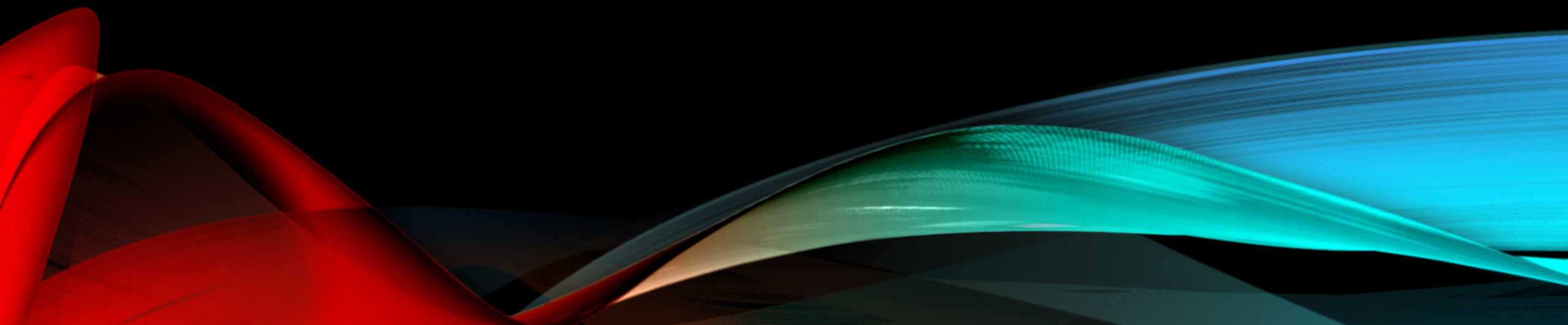
Telegram

Signal

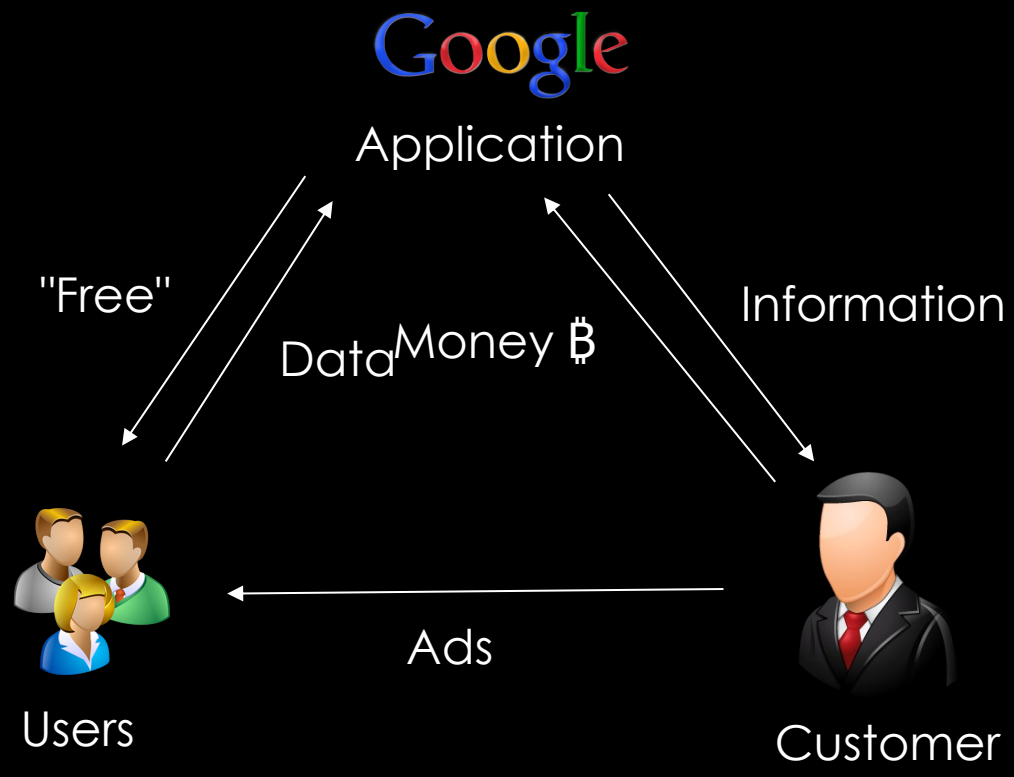
Thank you!
If you want to know more join our Telegram / Signal groups

ONE LAST THING

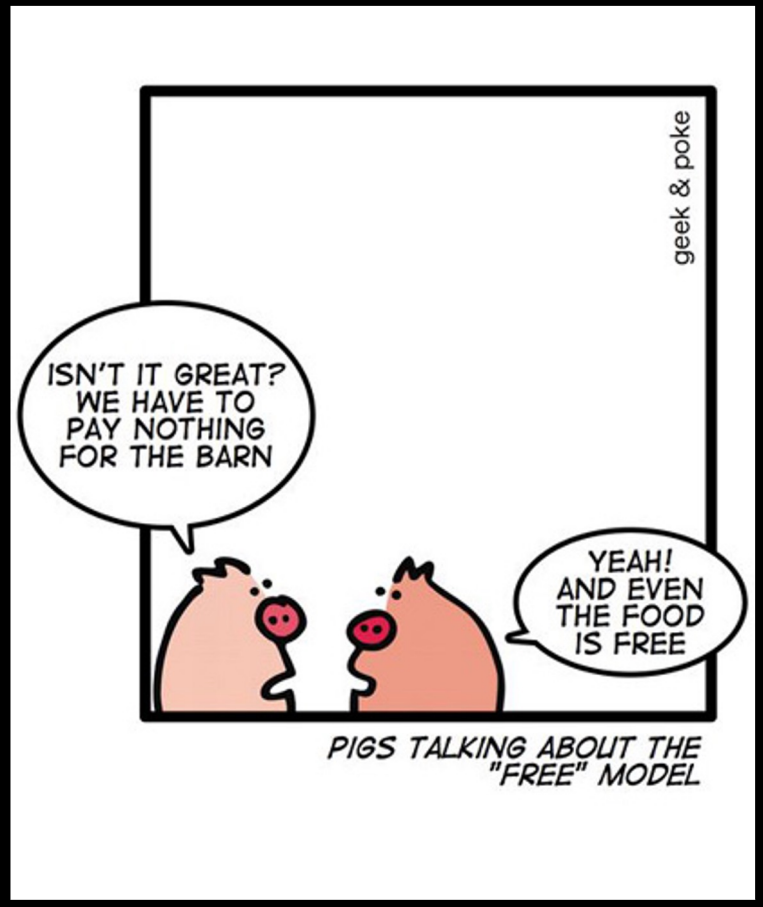
The elephant in the room



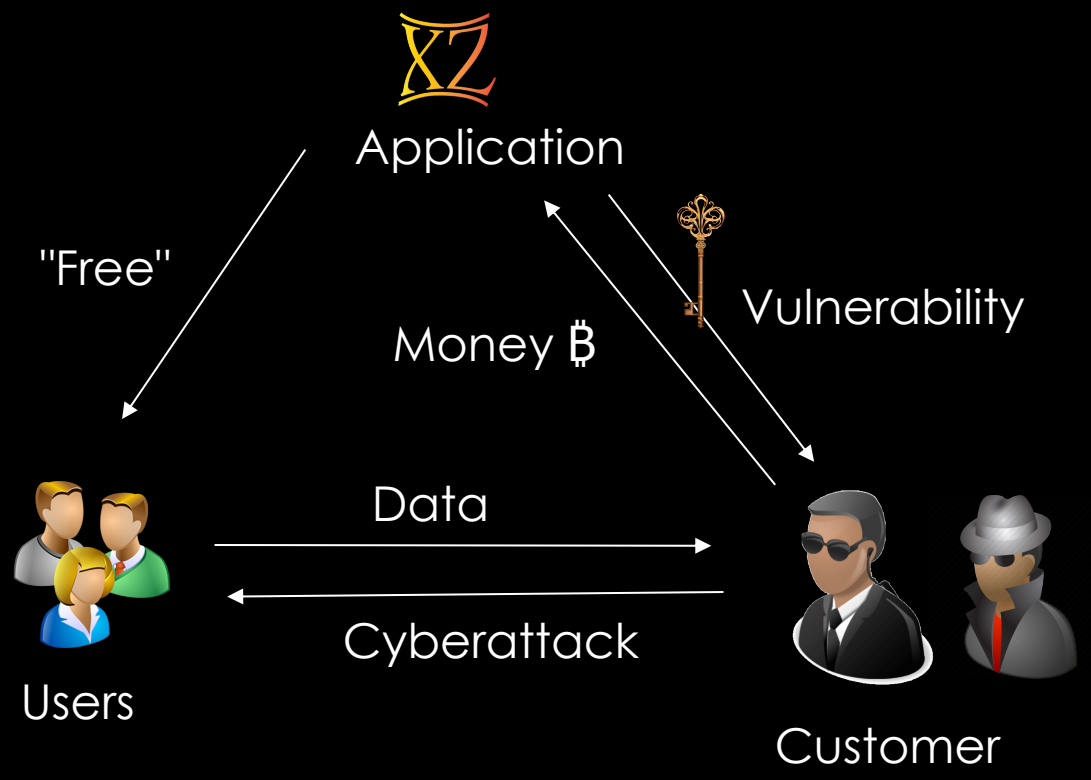
GOOGLE BUSINESS MODEL



THE “FREE” BEER SOFTWARE BUSINESS MODEL



JIA TAN BUSINESS MODEL



IS OPENSSH PRACTICING THE SEGELMAN BUSINESS MODEL?

- QR Code above leads to my blog, where my conclusions on the topic are presented.