Reducing The Surface Area Of Risk In Data Security Using Data Masking

Tim Gorman | Delphix | Senior Technical Manager

PostgresConf US 2018

Jersey City / United States

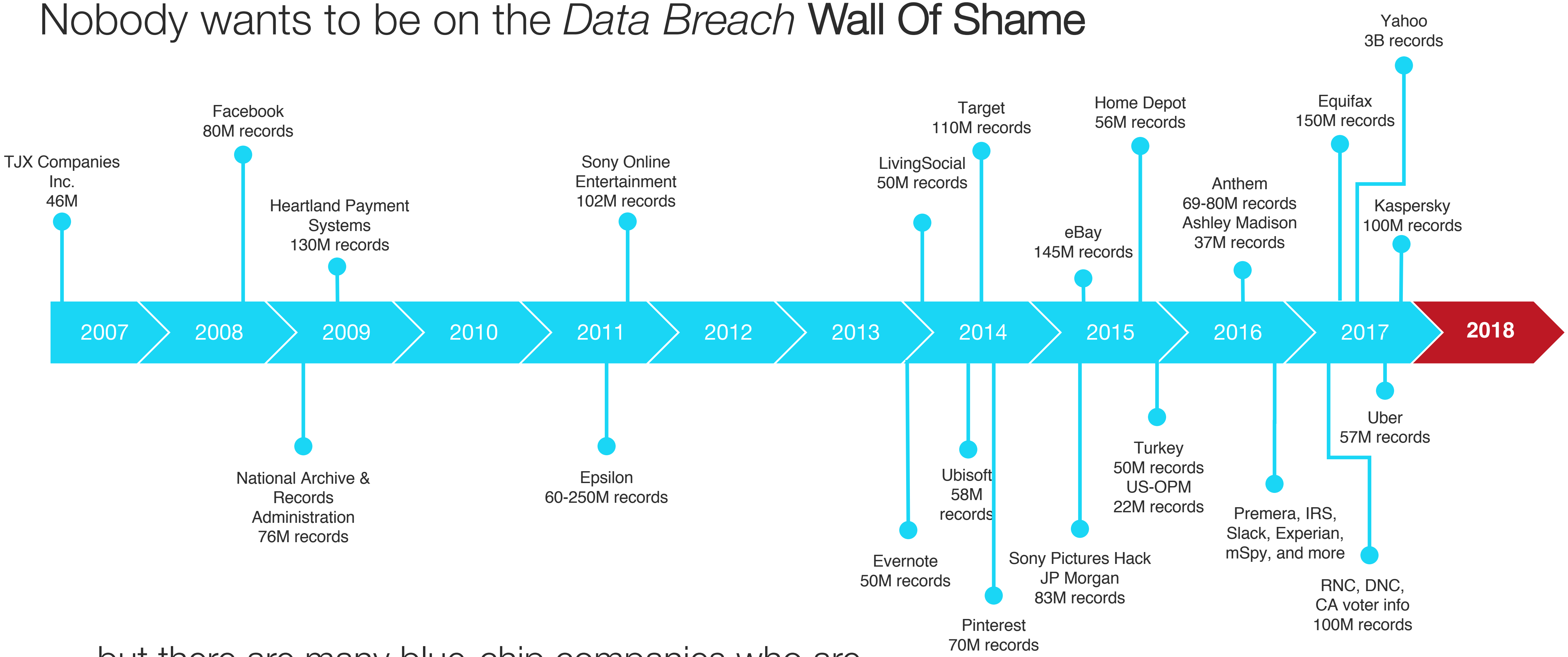
April 16 - 20, 2018

Agenda

1. **Fear and loathing**
2. External and internal threats
3. Data masking
4. Summary

Fear and Loathing

Nobody wants to be on the *Data Breach* **Wall Of Shame**



TJX Companies Inc. 46M
Facebook 80M records
Heartland Payment Systems 130M records
National Archive & Records Administration 76M records
Sony Online Entertainment 102M records
Epsilon 60-250M records
LivingSocial 50M records
Target 110M records
Evernote 50M records
Ubisoft 58M records
Pinterest 70M records
eBay 145M records
Home Depot 56M records
Sony Pictures Hack JP Morgan 83M records
Turkey 50M records US-OPM 22M records
Anthem 69-80M records Ashley Madison 37M records
Premera, IRS, Slack, Experian, mSpy, and more
Equifax 150M records
Yahoo 3B records
Kaspersky 100M records
Uber 57M records
RNC, DNC, CA voter info 100M records
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018

…but there are many blue-chip companies who are

Fear and Loathing

- The attitude of many is…

We have a firewall. We're good.

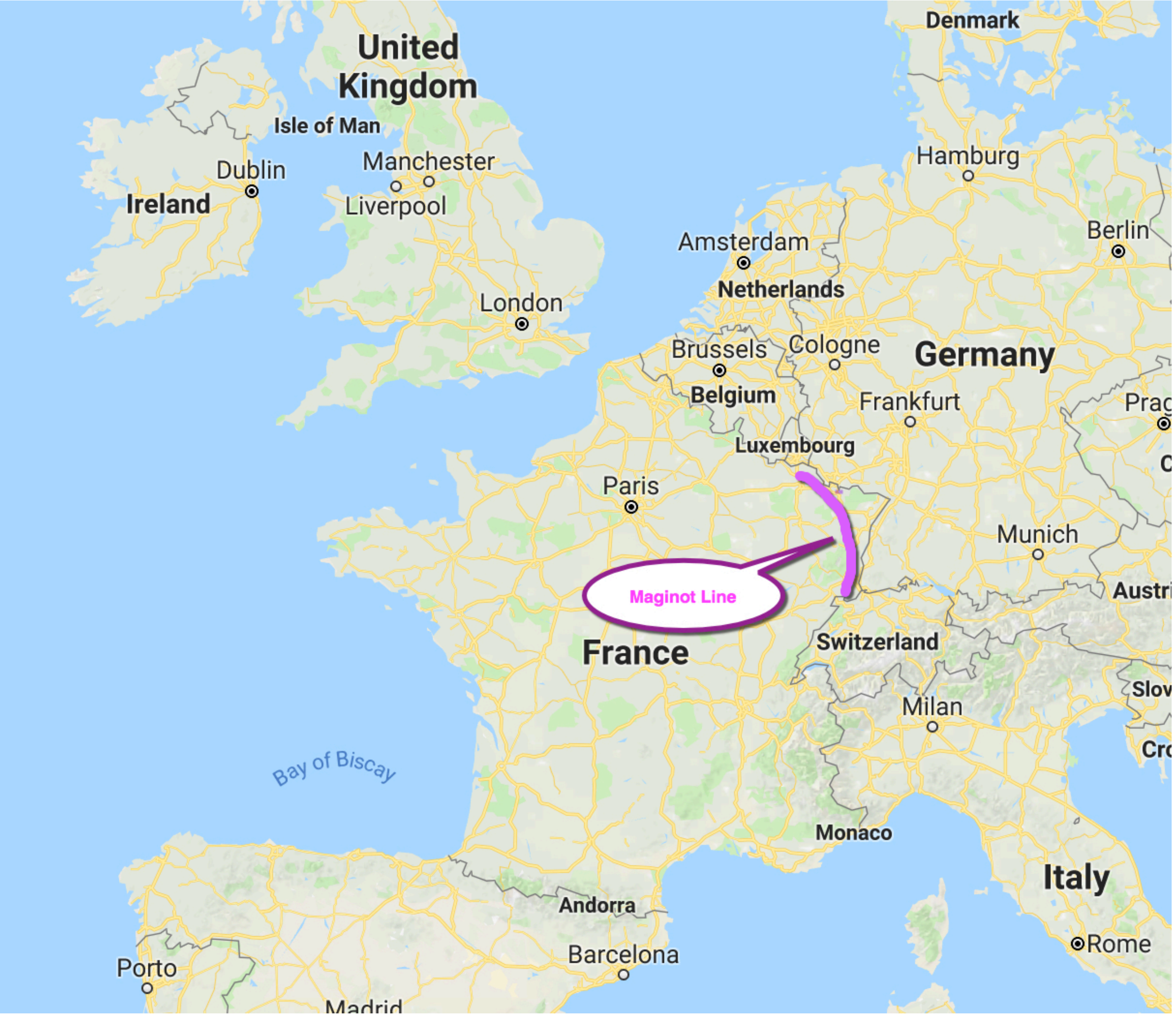
Tough luck for those other folks…

- In the 1930s, France built an enormous fortification known as the Maginot Line
  - It was designed specifically to prevent Germany from ever invading
  - Every military expert worldwide agreed that it was *impregnable*

Nous avons la ligne Maginot! Que peuvent faire les Boche?

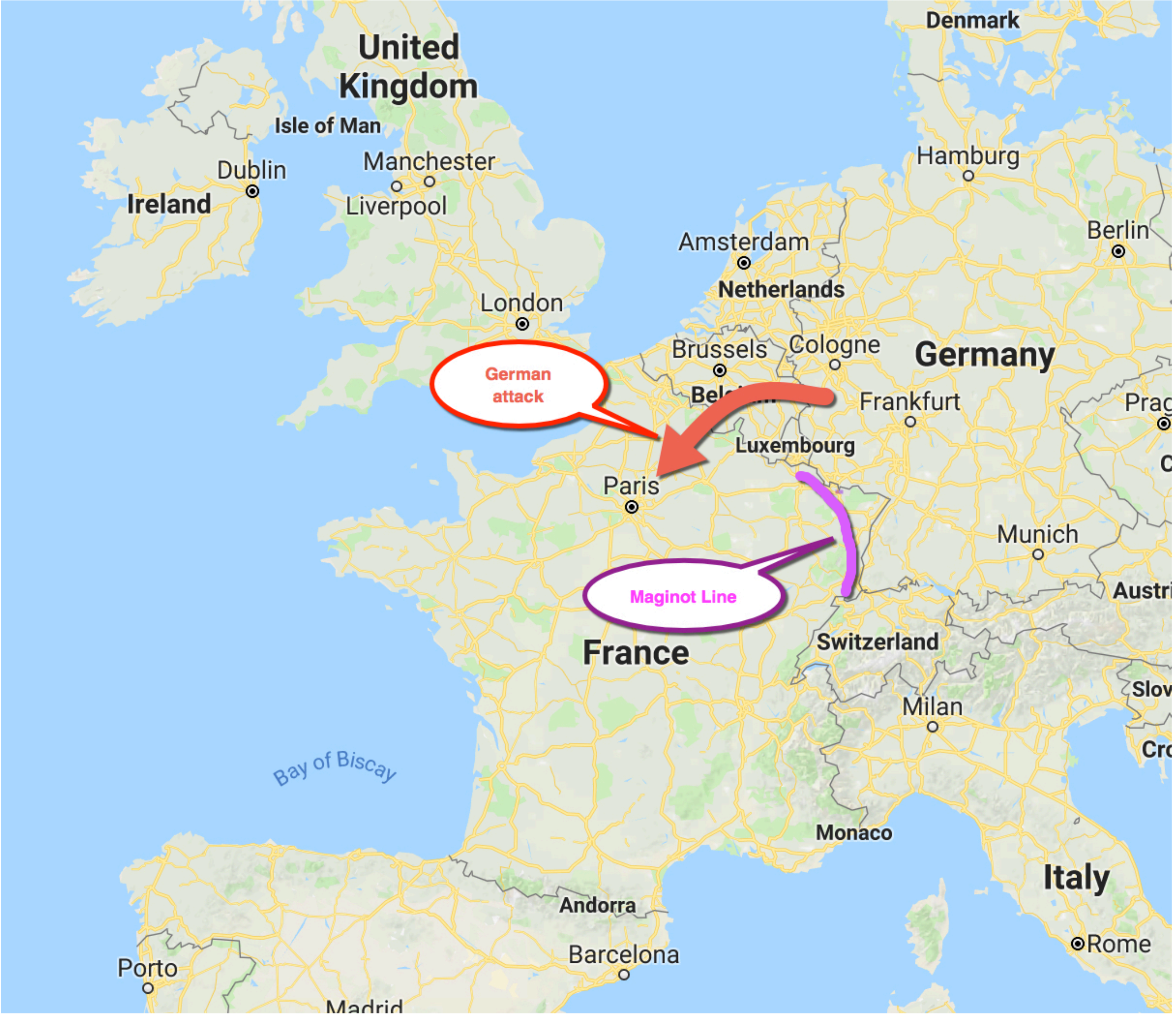
- In 1940, Germany conquered France in 6 weeks

Fear and Loathing



Fear and Loathing

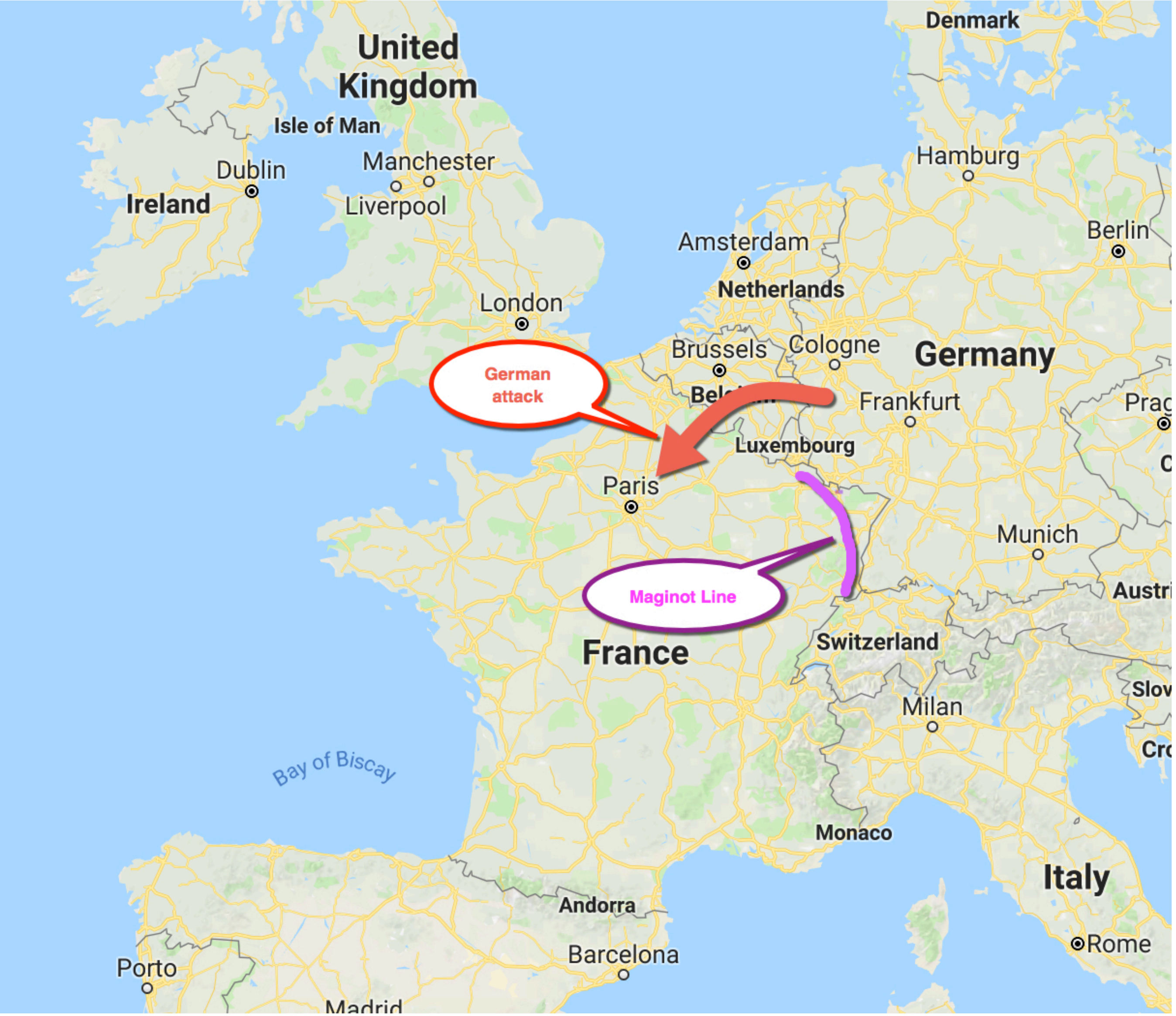
- Germany simply *bypassed* the Maginot Line and conquered France in 6 weeks



Fear and Loathing

- Germany simply *bypassed* the Maginot Line and conquered France in 6 weeks

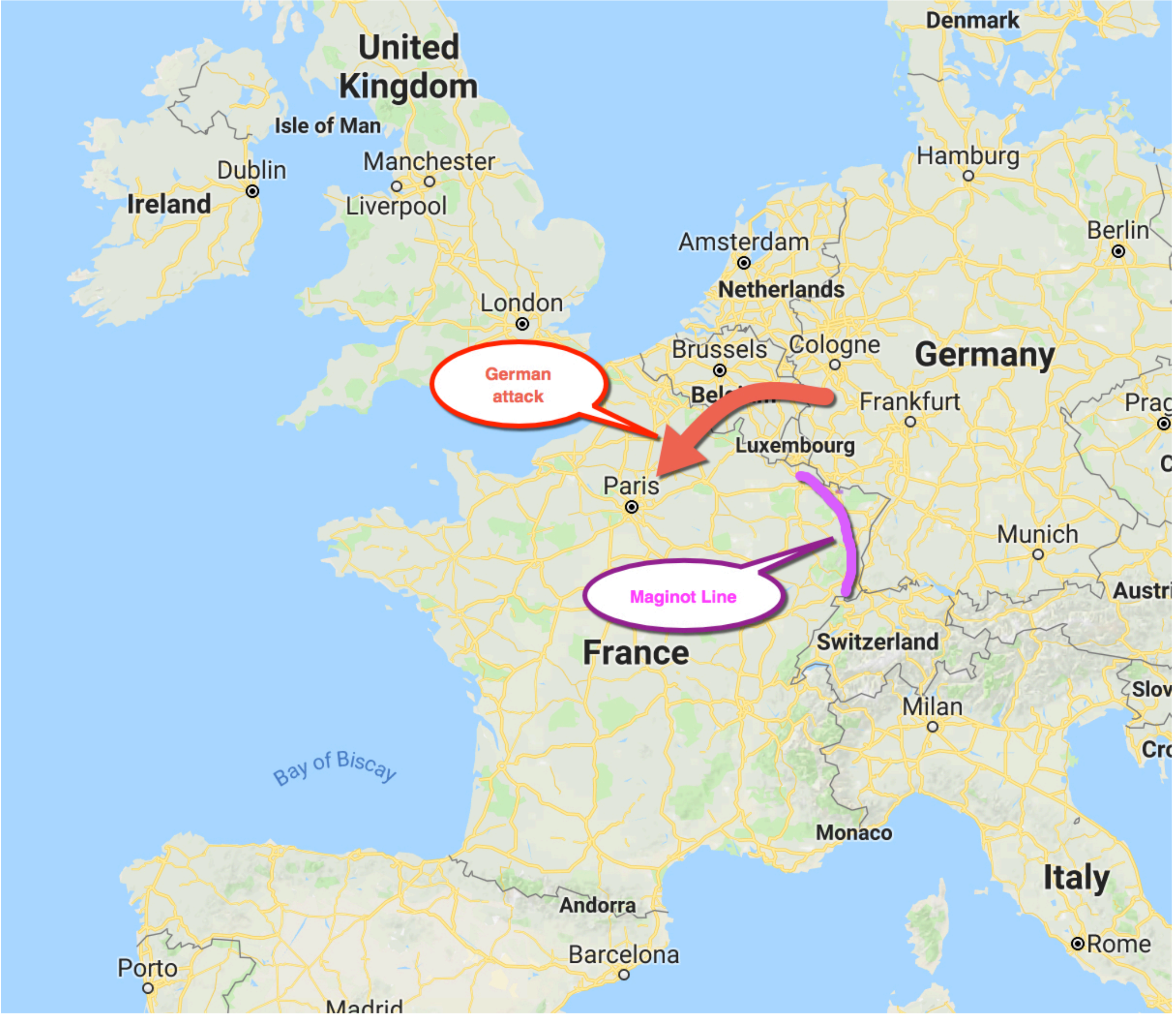
- *Lessons learned*:



Fear and Loathing

- Germany simply *bypassed* the Maginot Line and conquered France in 6 weeks

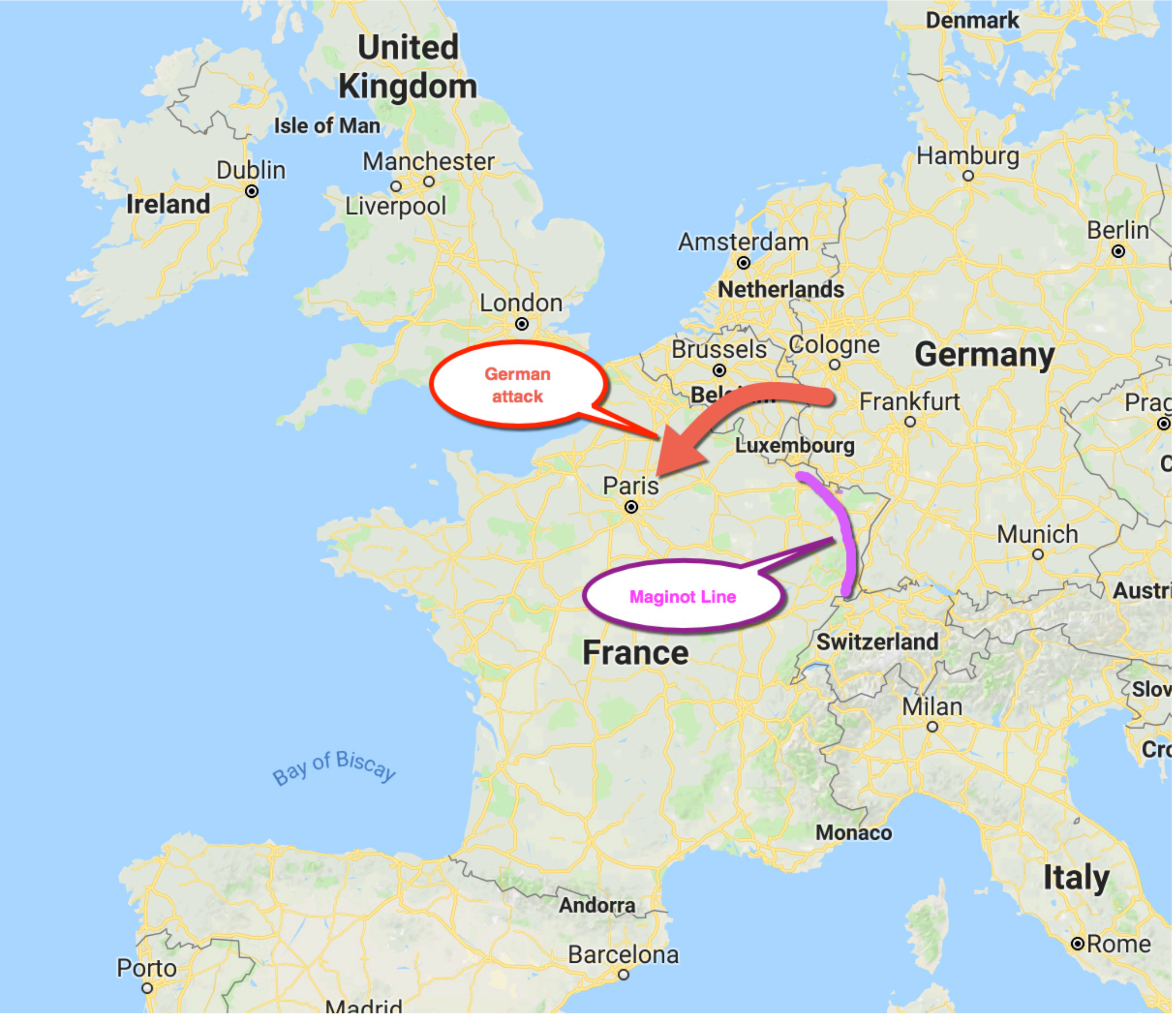
- *Lessons learned*:
  1. Use multiple layers of defense
     - *Do **not** rely on a single strong defense against a single threat*



Fear and Loathing

- Germany simply *bypassed* the Maginot Line and conquered France in 6 weeks

- *Lessons learned*:
  1. Use multiple layers of defense
     - *Do **not** rely on a single strong defense against a single threat*
  2. Create strongpoints and concentrate defenses within
     - *Impossible to defend **everything** equally, so **prioritize** and **focus***



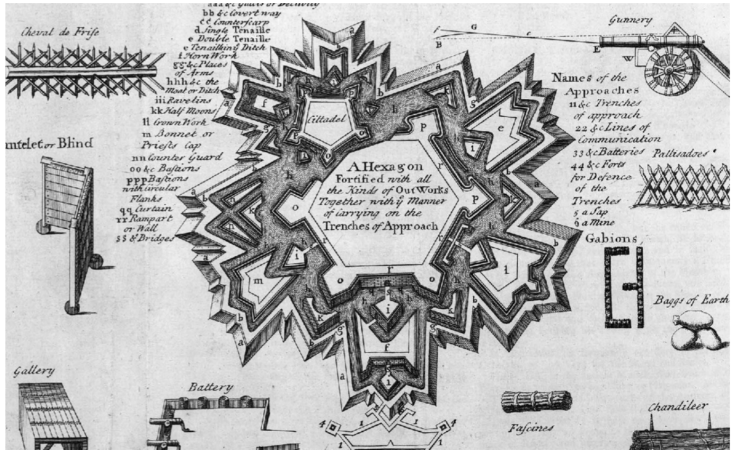
Agenda

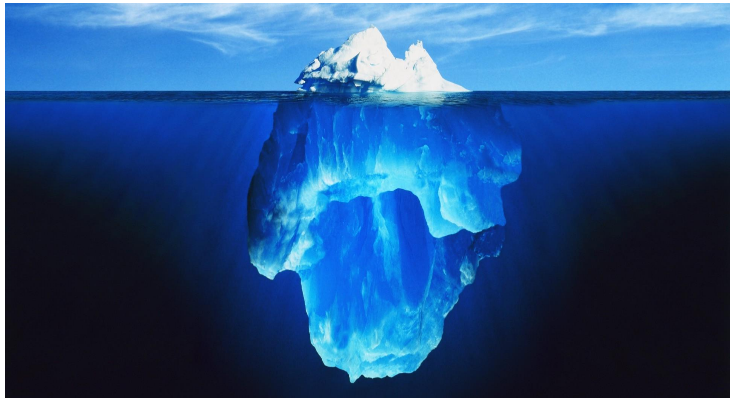
1. Fear and loathing
2. External and internal threats
3. Data masking
4. Summary

External and internal threats

How do we apply these lessons to prevent data breaches?

1. Layered defenses
   a) Physical security of data center
   b) Network security (firewalls)
   c) Strong authentication to services and servers
   d) Centralized rule-based authorization to services and servers
   e) Encryption of data in-flight
   f) Encryption of data at-rest
2. Reduce the surface area of risk
   a) Prioritize and focus protection efforts on production systems
   b) Mask (obfuscate) sensitive/confidential data in non-production systems

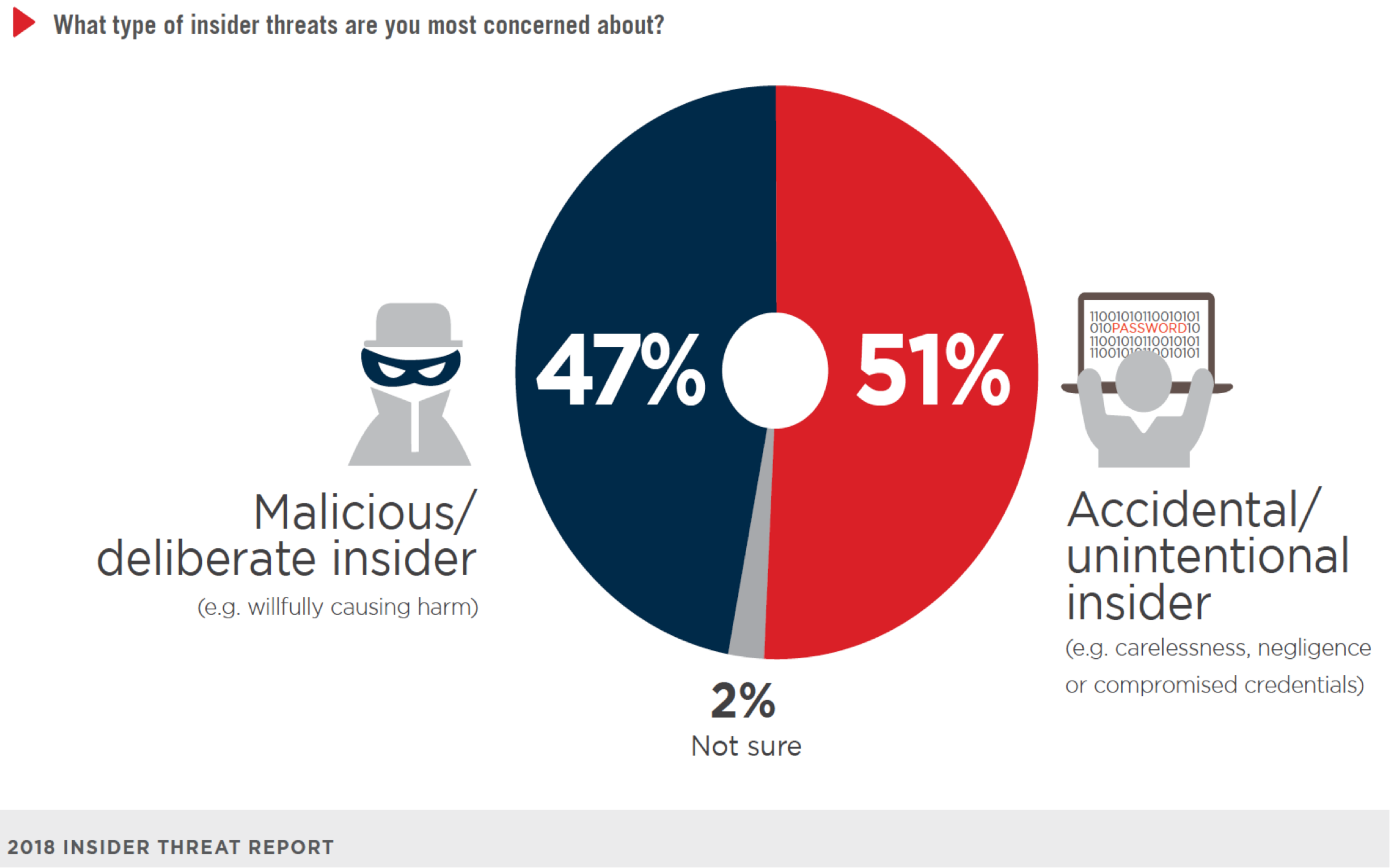




External and internal threats

In addition to attacks from **external vectors**, there is growing realization about the nature of **insider threats**

- 90% of organizations *feel vulnerable* to insider attack
  - The main enabling risk factors include…
    - too many users with excessive access privileges (37%)
    - an increasing number of devices with access to sensitive data (36%)
    - increasing complexity of information technology (35%)
- 53% *confirmed* insider attacks against their organization in the **previous 12 months**
  - Typically fewer than 5 attacks, but 27% say insider attacks have become more frequent

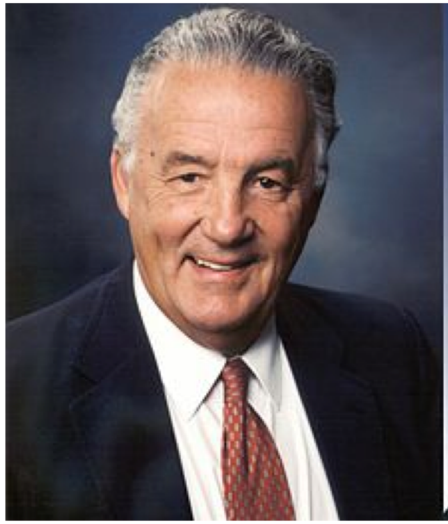


What type of insider threats are you most concerned about?

Courtesy of: *2018 Insider Threat Report – Cybersecurity-Insiders.com and Crowd Research Partners*

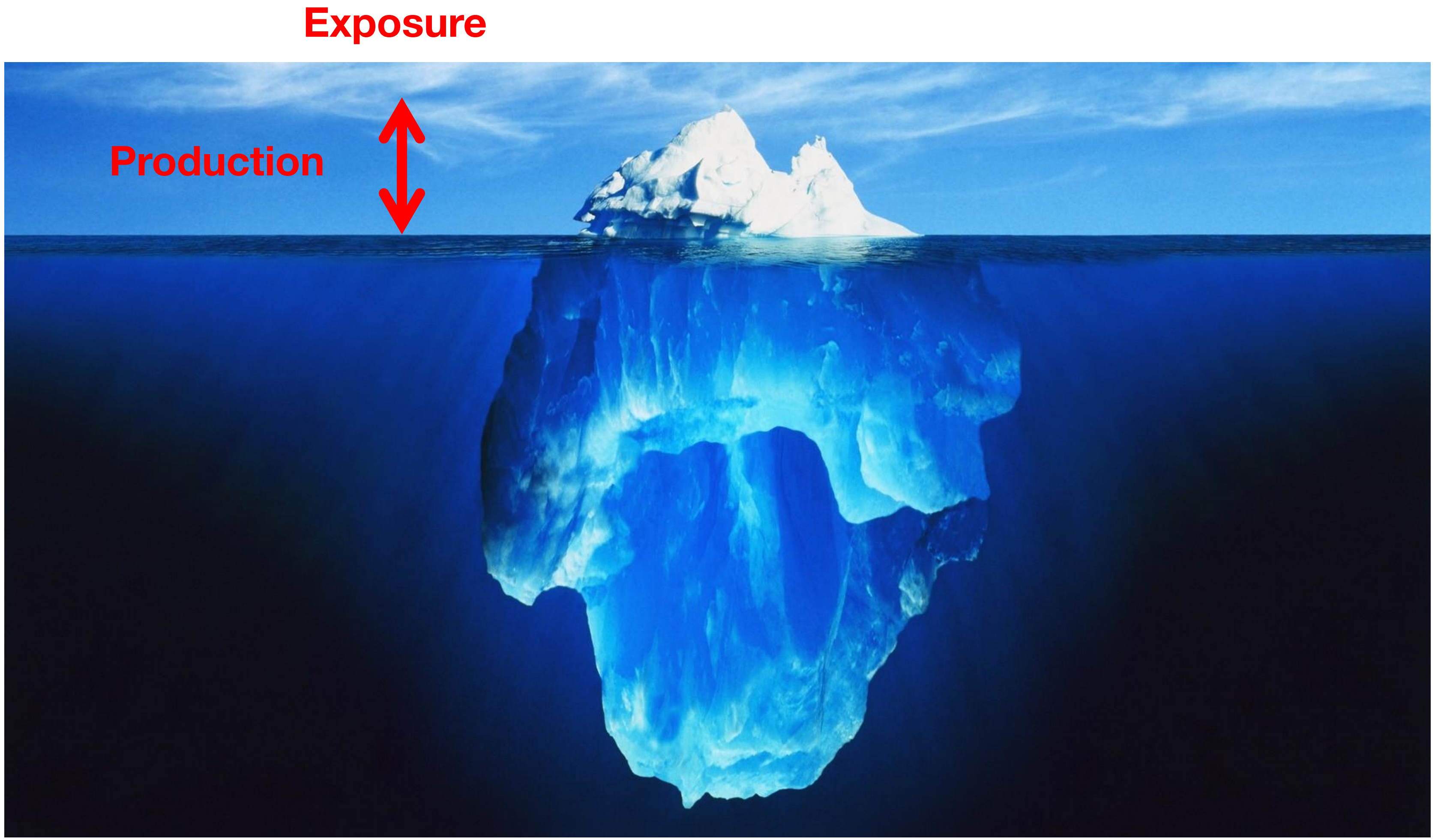
External and internal threats

- For decades, there had been an unspoken *honor code* in place in IT…
  - In the US, the Sarbanes-Oxley Act of 2002 forcefully brought attention to this potential liability by imposing penalties for corporate malfeasance on the CEO and CFO
    - Similar laws in many countries
  - The Snowden debacle in 2013 demonstrated how low-level IT staff could abuse the honor code to cause a breach
- Reliance on this *honor code* is a liability
  - On the organization
  - On IT personnel
  - GDPR in the EU is the final nail in the coffin
- How do we minimize the risks?



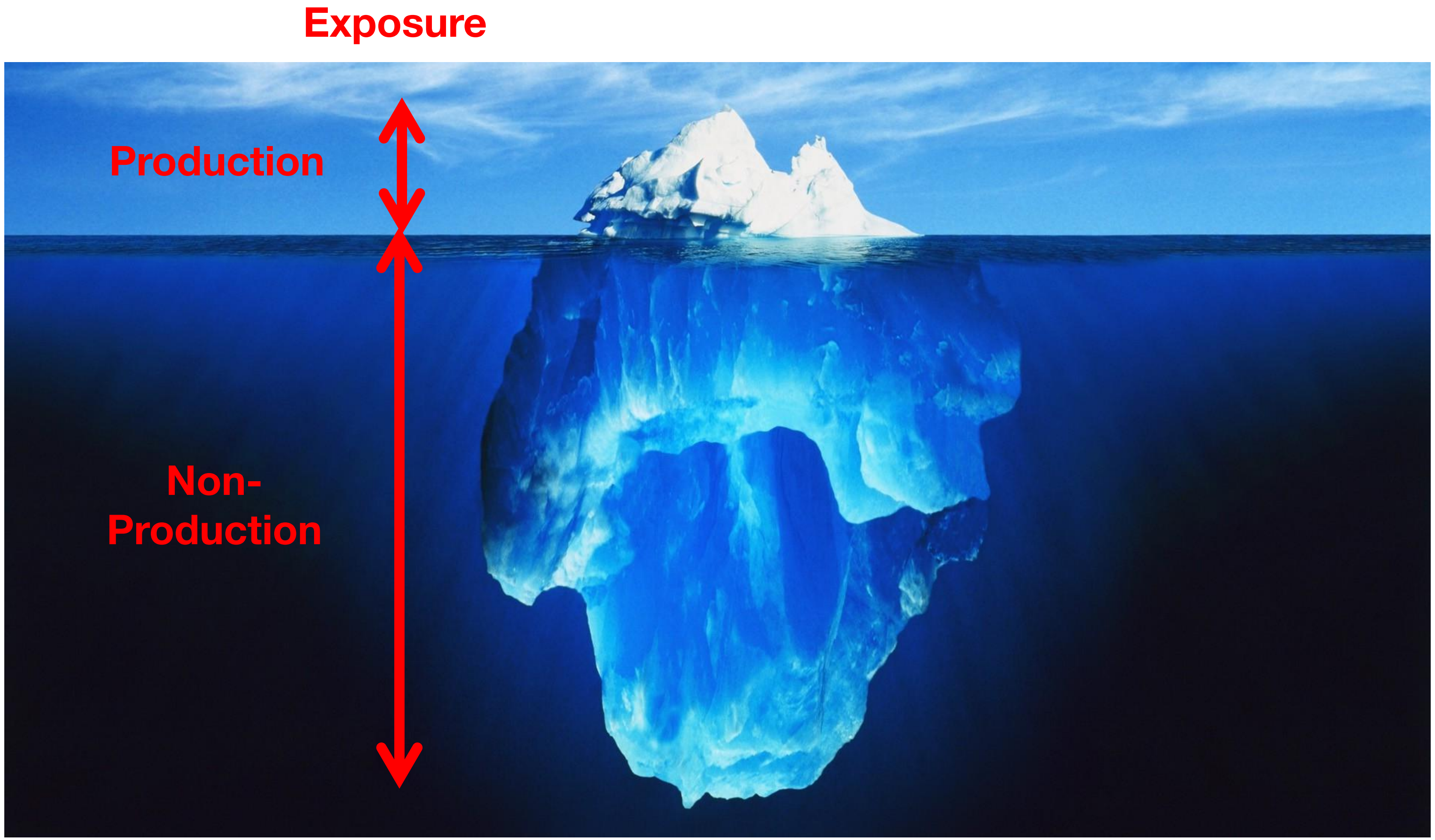
External and internal threats

- Non-production environments represent an enormous increase in the *surface area of risk* for exposure of sensitive production data



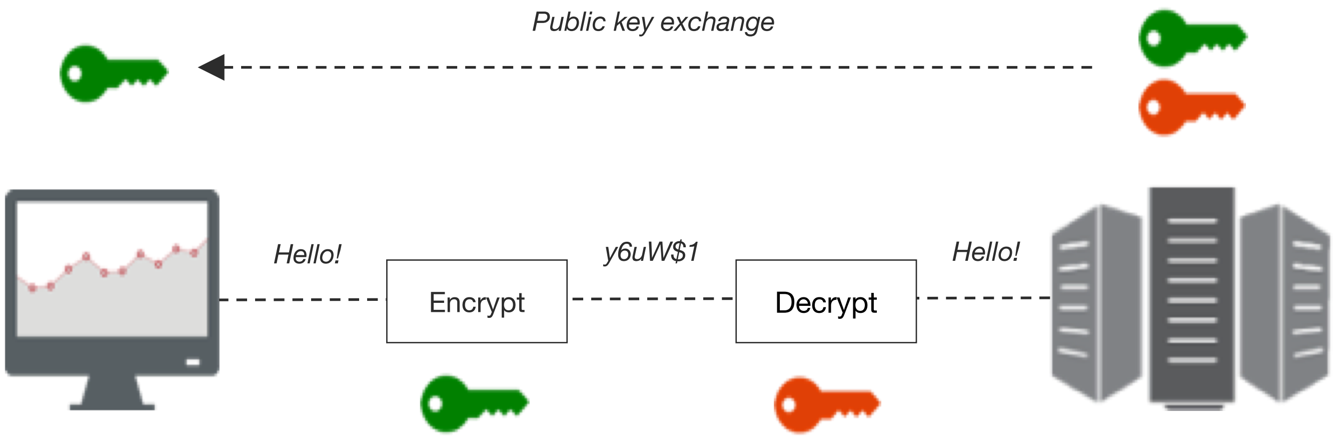
External and internal threats

- Non-production environments represent an enormous increase in the *surface area of risk* for exposure of sensitive production data



External and internal threats

- **Encryption** is the process of encoding data in such a way that only authenticated and authorized parties can decrypt it
- Decryption = ***reversible*** obfuscation



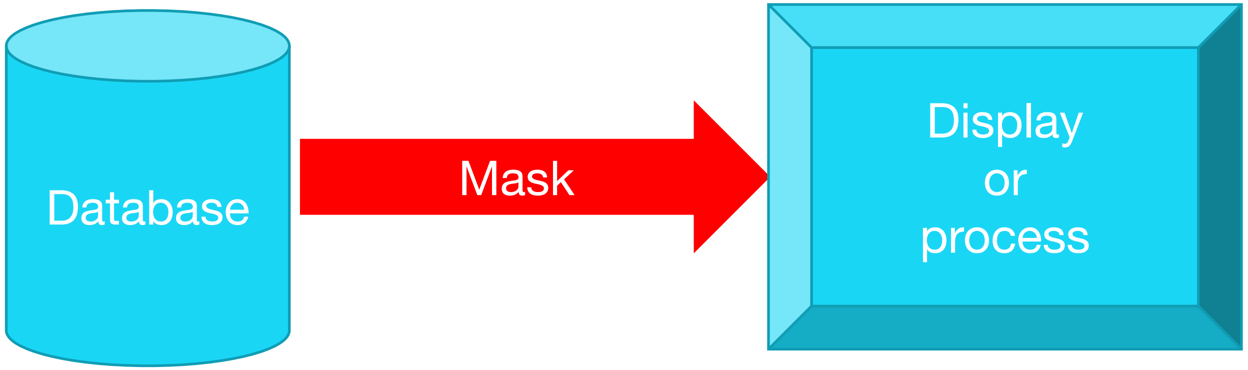
- In non-production, developers and testers must be authorized to decrypt data to do their jobs
- What if they aren’t really authorized to view sensitive data?

ADVANTAGES

- Effective for sending data such as emails or files between two secured locations (*data in-flight*)
- Effective for protecting data in a production application (*data at-rest*)

External and internal threats

- **Masking data in-flight** is the obfuscation of data **after** it has been retrieved from storage **at-rest**
- Masking = ***non-reversible*** obfuscation



ADVANTAGES

- Effective for obfuscating data in production systems by not changing data at-rest

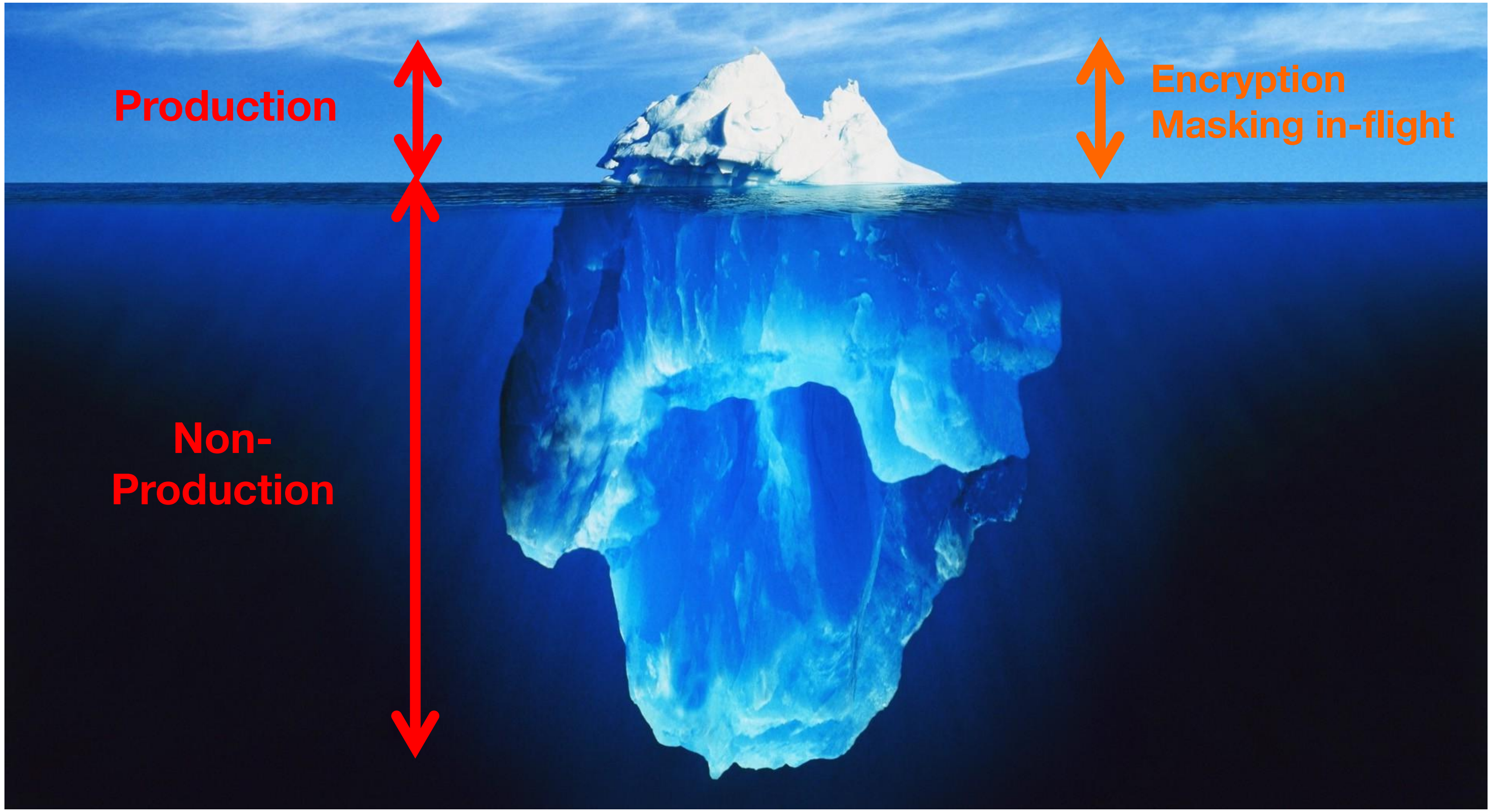
- SQL Server Dynamic Data Masking (DDM) is an example

External and internal threats

- **Encryption** is the appropriate solution in **production** systems
  - *obfuscation* which is *reversible* upon *authorization*

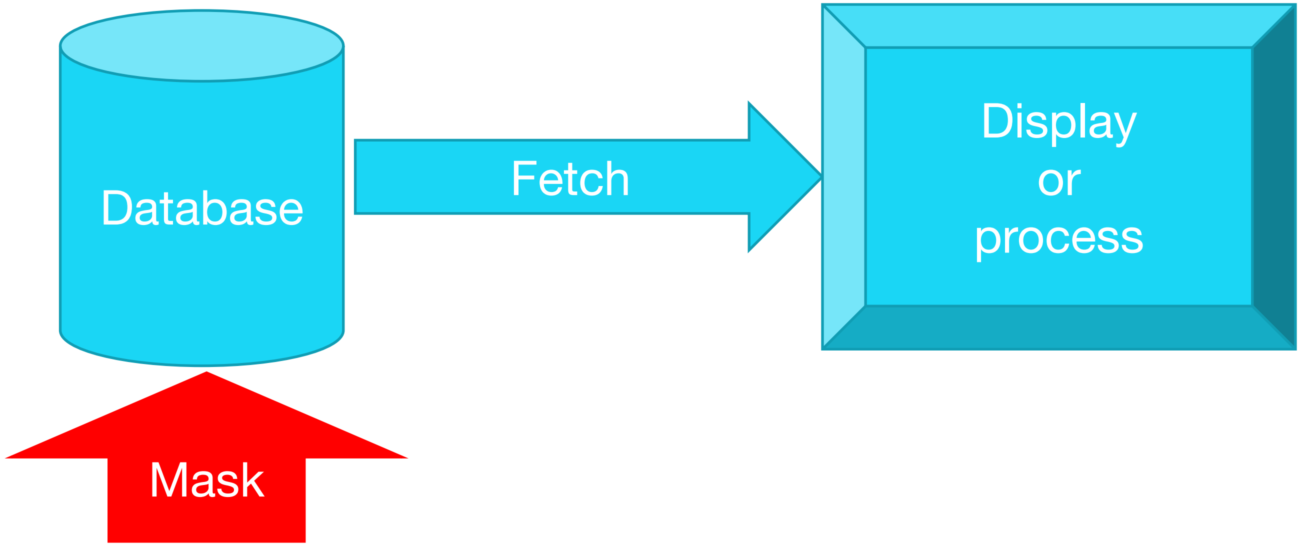
Exposure

Solution



External and internal threats

- **Masking data at-rest** is the obfuscation of data within the database using SQL statements
- Masking = ***non-reversible*** obfuscation



ADVANTAGES

- Effective for obfuscating data in non-production systems by changing data at-rest
- Allows provisioning non-production systems outside of secured authorized environments

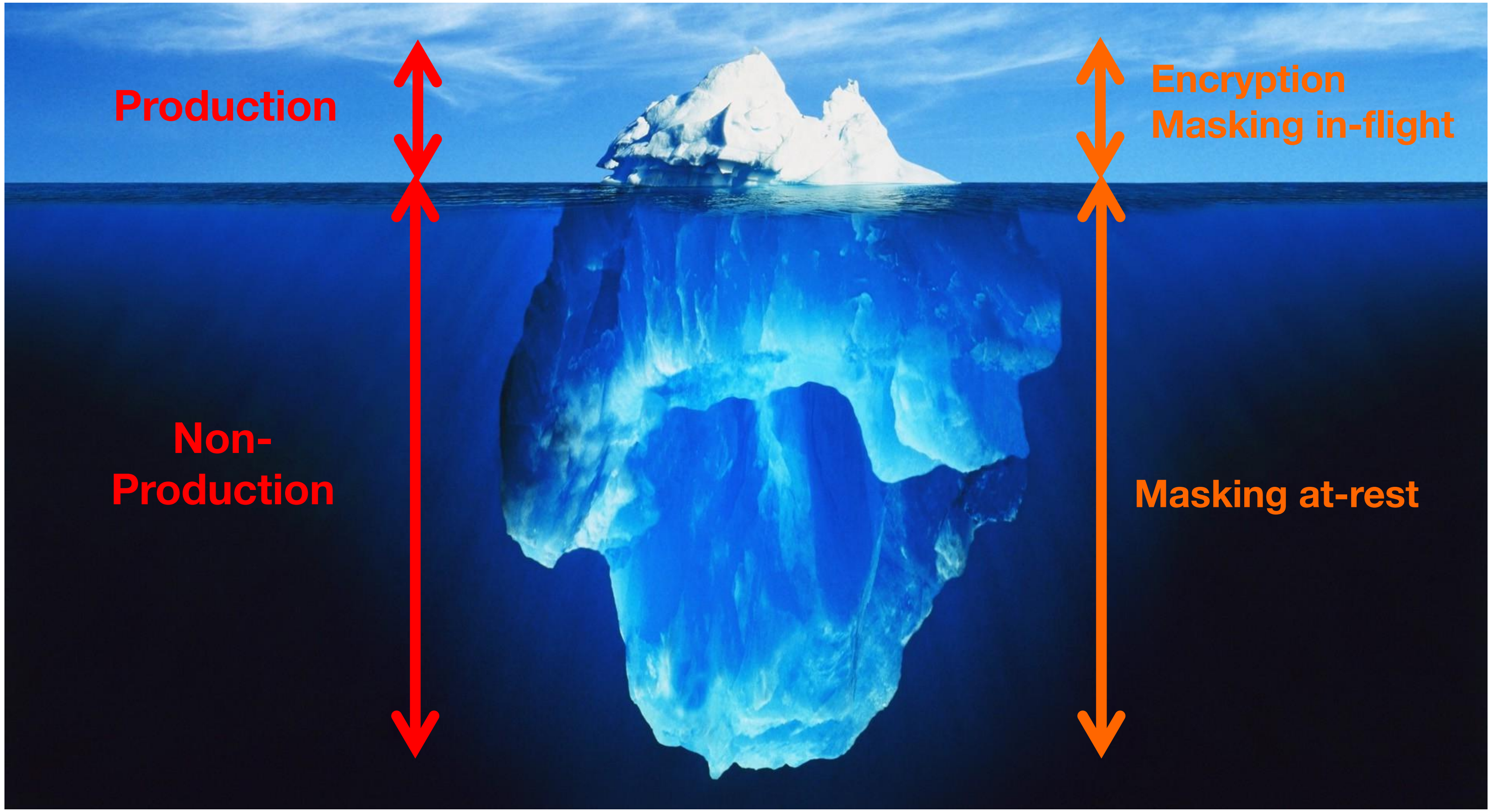
- Delphix, IBM Optim, Informatica data masking are examples

External and internal threats

- **Encryption** and **masking in-flight** are appropriate solutions in **production** systems
  - *obfuscation* which is *reversible* upon *authorization*
- **Masking at-rest** is the appropriate solution in **non-production** systems
  - *obfuscation* which is *never reversible*

Exposure

Solution



External and internal threats

- Database virtualization
  - For decades, non-production databases have been created using…
    - Database copies from production
    - Newly-created databases with generated data
  - Data virtualization technologies are now available
    - Thin-clone copies of databases sourced from production presented via network-attached storage
    - Allows DBAs to create TB-sized database copies in less than 10 minutes
      - Delphix, Windocks, Red Gate, Rubrik, Actifio, etc
  - So, by cloning production to create dozens or hundreds of copies for non-production…

…somewhere a security administrator is writhing in agony

Agenda

1. Fear and loathing
2. External and internal threats
3. **Data masking at-rest**
4. Summary

Data masking

1
- Masking at-rest must not be reversible

2
- The results must be representative of the data source

3
- Referential integrity must be maintained

4
- Only mask non-sensitive data if it can be used to infer sensitive data

5
- Masking must be a repeatable process

According to Rich Mogull, Securosis

Data masking

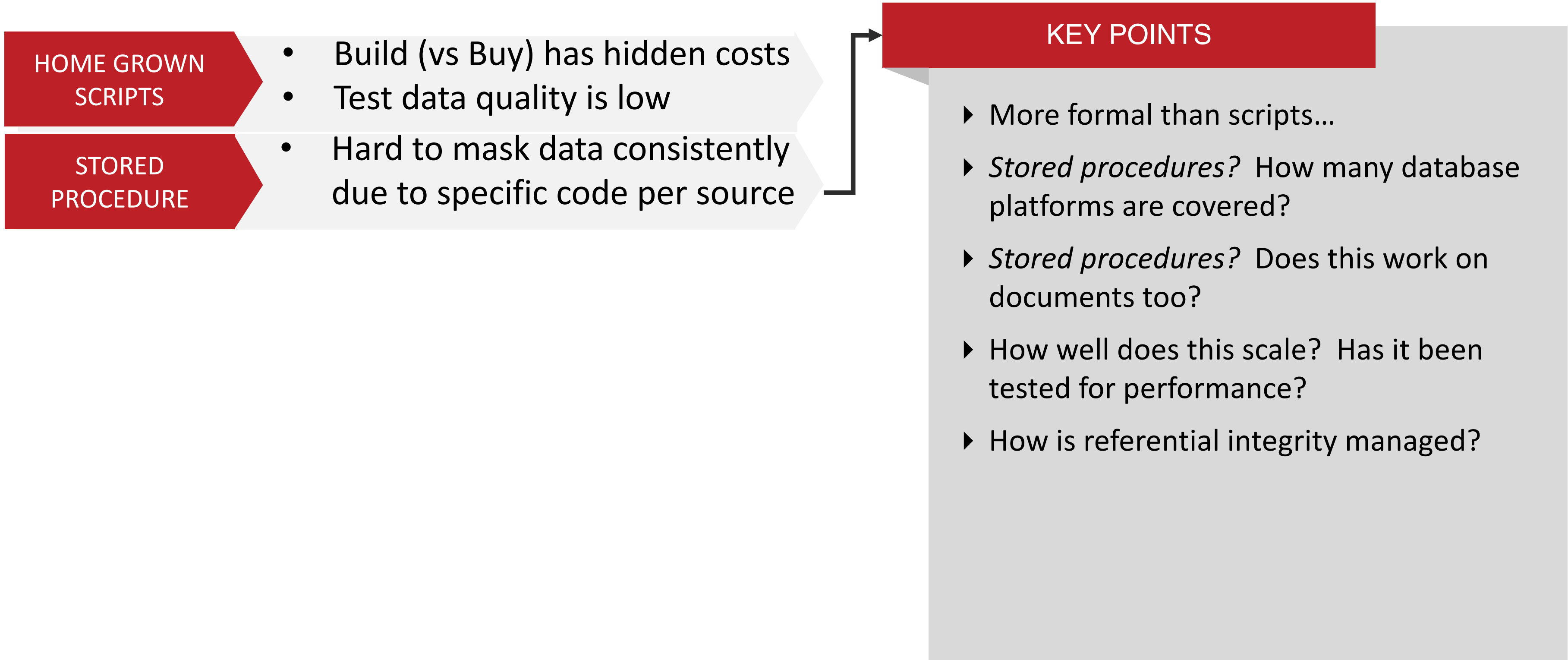
HOME GROWN SCRIPTS

- Build (vs Buy) has hidden costs
- Test data quality is low

KEY POINTS

- How much in-house expertise do we really have for obfuscating data?
- Are the scripts reusable? Across database platform? Across platform? Across all documents?
- How well has it been tested?
- Are we really serious about protecting confidential data? Or just checking off an item from a task list?

Data masking



HOME GROWN SCRIPTS

- Build (vs Buy) has hidden costs
- Test data quality is low

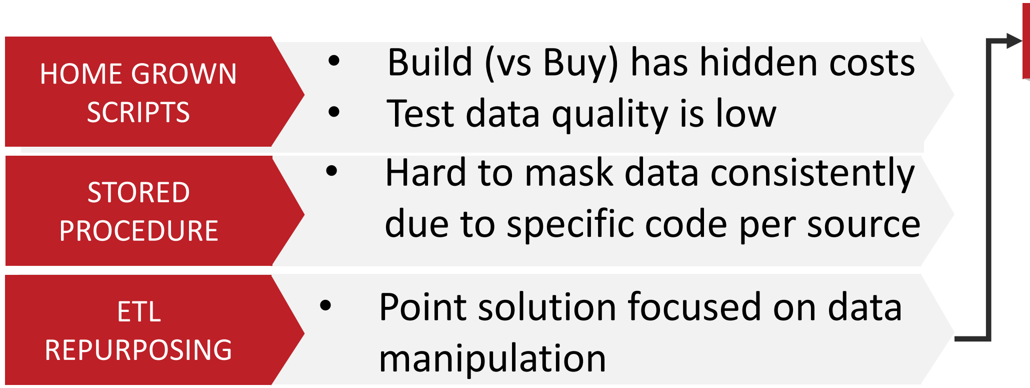
STORED PROCEDURE

- Hard to mask data consistently due to specific code per source

KEY POINTS

- More formal than scripts…
- *Stored procedures?* How many database platforms are covered?
- *Stored procedures?* Does this work on documents too?
- How well does this scale? Has it been tested for performance?
- How is referential integrity managed?

Data masking



HOME GROWN SCRIPTS

- Build (vs Buy) has hidden costs
- Test data quality is low

STORED PROCEDURE

- Hard to mask data consistently due to specific code per source

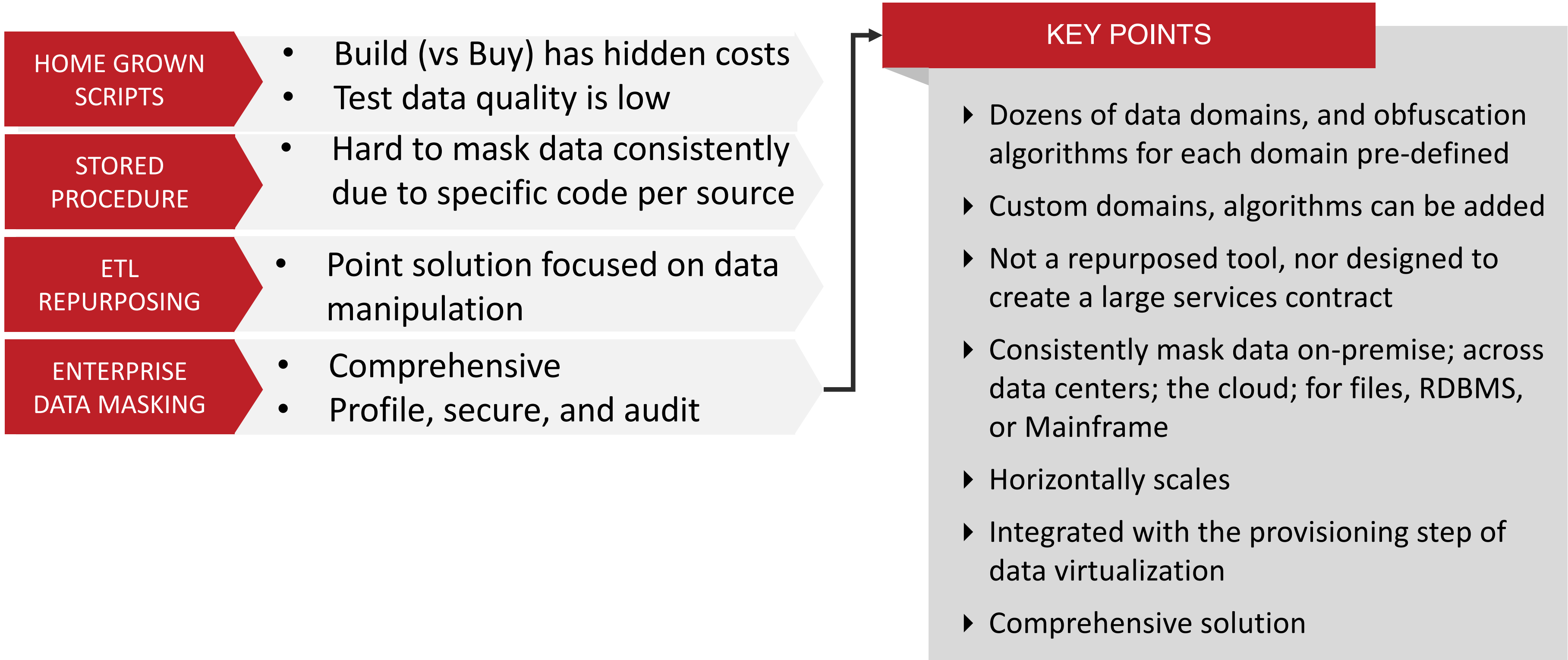
ETL REPURPOSING

- Point solution focused on data manipulation

KEY POINTS

- Obfuscating data is about *extracting* unmasked data, *transforming* data, and then *loading* masked data back, so this seems a natural evolution, but…
- Is there a "discovery" function embedded to ensure that all confidential data is identified?
- Is there an "audit" function to track when data was audited and if new data has been added since?

Data masking



HOME GROWN SCRIPTS

- Build (vs Buy) has hidden costs
- Test data quality is low

STORED PROCEDURE

- Hard to mask data consistently due to specific code per source

ETL REPURPOSING

- Point solution focused on data manipulation

ENTERPRISE DATA MASKING

- Comprehensive
- Profile, secure, and audit

KEY POINTS

- Dozens of data domains, and obfuscation algorithms for each domain pre-defined
- Custom domains, algorithms can be added
- Not a repurposed tool, nor designed to create a large services contract
- Consistently mask data on-premise; across data centers; the cloud; for files, RDBMS, or Mainframe
- Horizontally scales
- Integrated with the provisioning step of data virtualization
- Comprehensive solution

Data masking

- Secure Lookup Algorithm
  - One of eight (8) data transformation frameworks pre-built into the Delphix masking engine
    - Patented proprietary encrypt / hash / modulus lookup algorithm, repeatable yet unbreakable
  - Used to assign a realistic value from a value selected from a pre-defined lookup table
    - The algorithm is irreversible and purposely creates collisions in the output values for added security
- Example
  1. Starting with original column value of “`XYZ Holdings`”
  2. original table has about 1000 distinct data values in the column
     - lookup table can be defined with 500 distinct data values
  3. Encrypt original value using AES 256 to “*`lGqll59bm7aX2C3bBVMJ3uIg%=`*”
  4. MD5 Hash of the encrypted result = “`428618117`”
  5. 428618117 mod 500 = `117`
  6. Value within lookup table at entry 117 is “`Standard Oil`”

Agenda

1. Fear and loathing
2. External and internal threats
3. Data masking
4. **Summary**

Summary

1. Understand the different choices for data security and their use-cases…
   1. **Encryption** and **masking in-flight** are good obfuscation solutions for **production** environments
      - Where all users are authenticated and authorized by the application
      - Where sensitive data can only be temporarily obfuscated
   2. **Data masking at-rest** is the right solution for non-production environments
      - Irreversibly make sensitive data *inconsequential* from a security perspective
      - Remove the value from the asset
2. Data masking at-rest products…
   - Delphix DMSuite, IBM Optim, Informatica Data Masking, Red Gate Data Masker, and more…
3. Job titles/descriptions that didn't exist in 2016 or 2017…
   - Data masking specialist
   - Data protection and vulnerability management specialist

…but they exist now and they're going to be important going forward…

Q & A

Tim.Gorman@Delphix.com

@TimGormanTech

aws

Pivotal

PostgresConf US 2018

Jersey City / United States

April 16 - 20, 2018