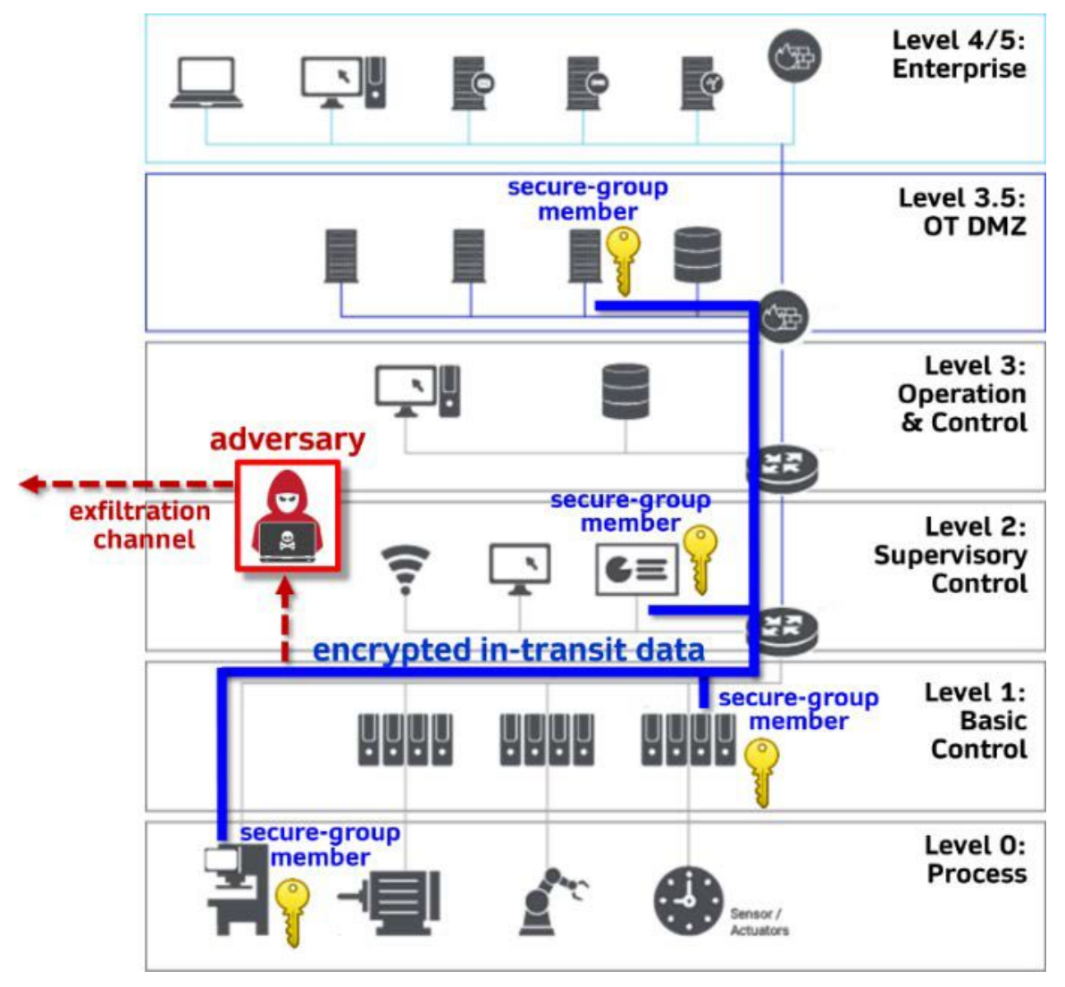
Risk-Based Optimization of Encryption Cryptoperiods for More Secure ICSs

Gabriele Cianfarani Natalija Vlajic

Department of Electrical Engineering and Computer Science, York University, Toronto, Canada

DATA SIPHONING

Industrial Control Systems (ICS) play a vital role in managing industrial processes efficiently and safely. Environments contain a **multitude of components** which need to communicate with one another. This increase in interconnectivity means that **secure** and **efficient** communication is critical.

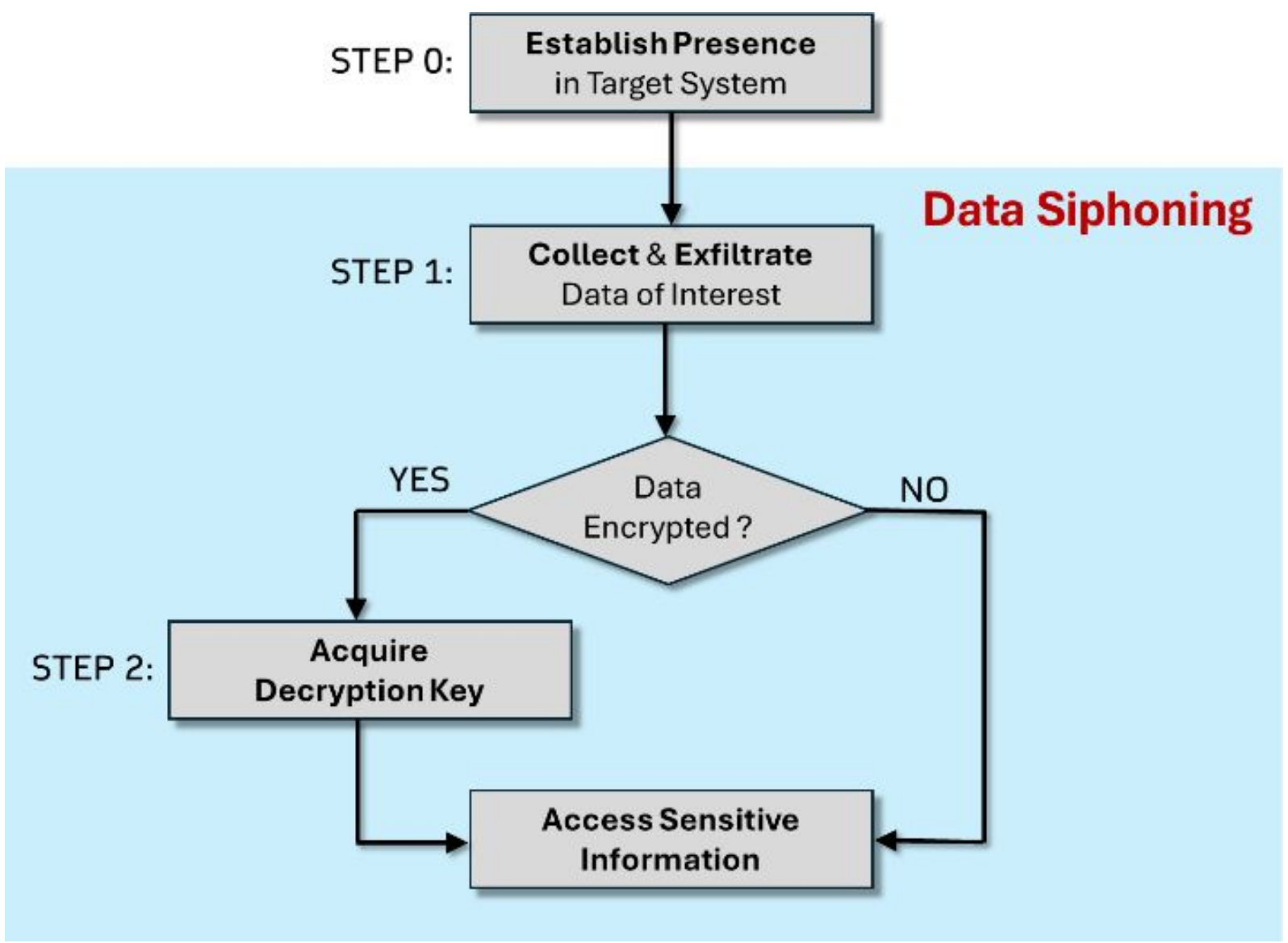


What is Data Exfiltration?

Internal network and **system reconnaissance** is one of the first crucial stages of most cyber attacks

- Involves the **unauthorized transfer** *(exfiltration)* of important operational **in-transit data**
- many modern ICSs use **encryption**, preventing immediate adversary access to exfiltrated data

Exfiltration must be accompanied with the **acquisition** of the respective ***encryption key***.


STEP 0:
Establish Presence in Target System
Data Siphoning
STEP 1:
Collect & Exfiltrate Data of Interest
Data Encrypted ?
YES
NO
STEP 2:
Acquire Decryption Key
Access Sensitive Information

What is Data Siphoning?

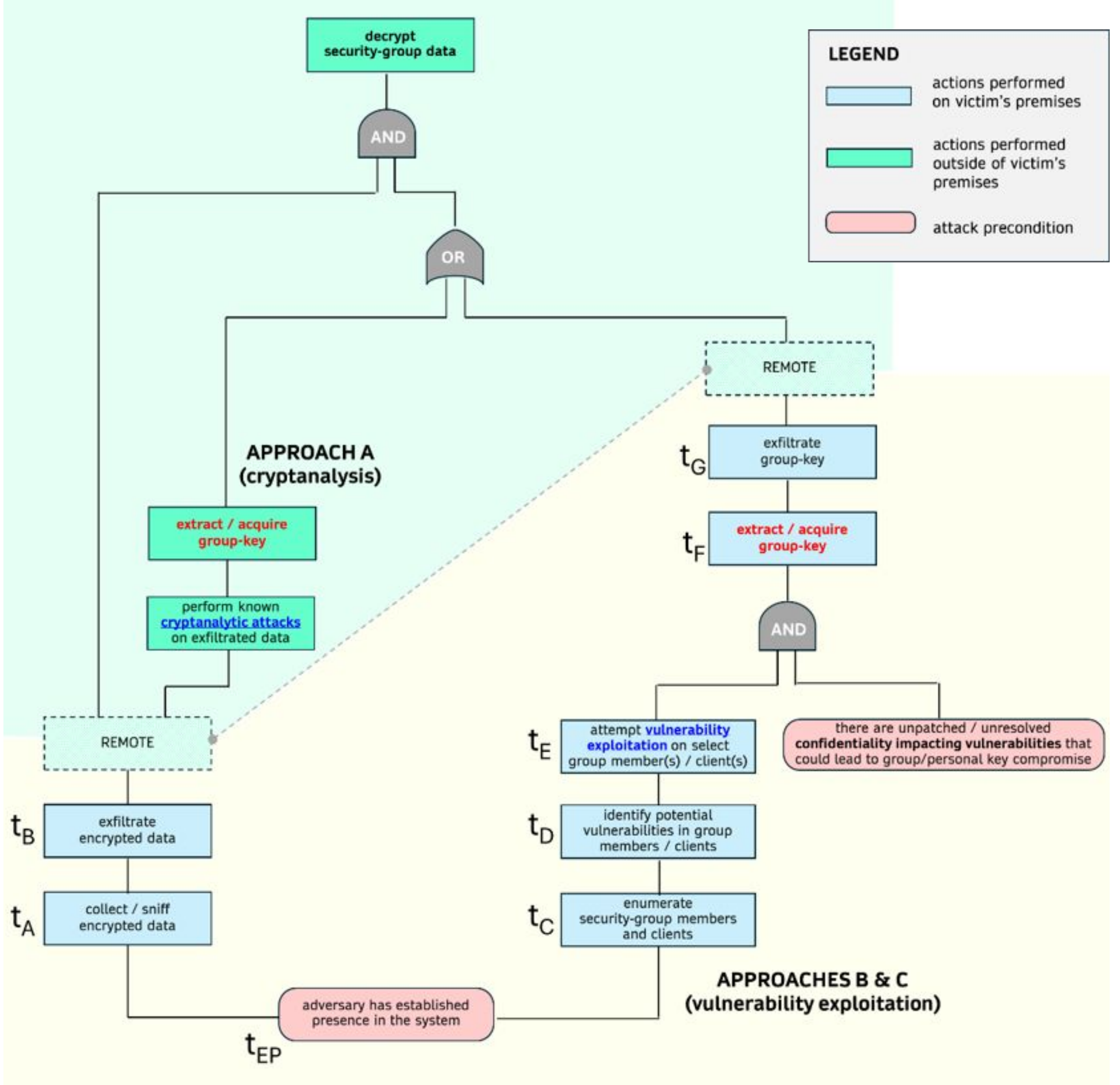
Data Siphoning refers to the **entire sequence of steps** that an adversary needs to perform to disclose certain **sensitive information** of interest.

STEP 1) collection and **exfiltration** of the given in-transit data

STEP 2) efforts to acquire the respective **decryption key** if/when the data from *STEP 1* happens to be encrypted

ATTACK TREE

An attack tree is a **bottom-up** diagram providing a bird's-eye view of the **logic** and **structure** of an attack.


decrypt security-group data
AND
OR
LEGEND
actions performed on victim's premises
actions performed outside of victim's premises
attack precondition
REMOTE
APPROACH A (cryptanalysis)
extract / acquire group-key
perform known cryptanalytic attacks on exfiltrated data
REMOTE
t_B
exfiltrate encrypted data
t_A
collect / sniff encrypted data
t_G
exfiltrate group-key
t_F
extract / acquire group-key
AND
t_E
attempt vulnerability exploitation on select group member(s) / client(s)
there are unpatched / unresolved confidentiality impacting vulnerabilities that could lead to group/personal key compromise
t_D
identify potential vulnerabilities in group members / clients
t_C
enumerate security-group members and clients
adversary has established presence in the system
t_{EP}
APPROACHES B & C (vulnerability exploitation)

What are the main methods of attack?

- **Approach A:** Perform **cryptanalysis** on the siphoned data
- **Approach B:** Gain control of a device through **exploitation** of a:
  - **software vulnerability**
  - **procedural vulnerability**
- **Approach C:** Obtain the key from the Security Key Service *(SKS)* by **impersonating** a legitimate client

Approach A generally takes place **outside** the victim's system, while **B and C** must be done **on-site**.

To minimize the impact of Data Siphoning attacks, various **risk-** and **loss-** minimizing measures can be deployed

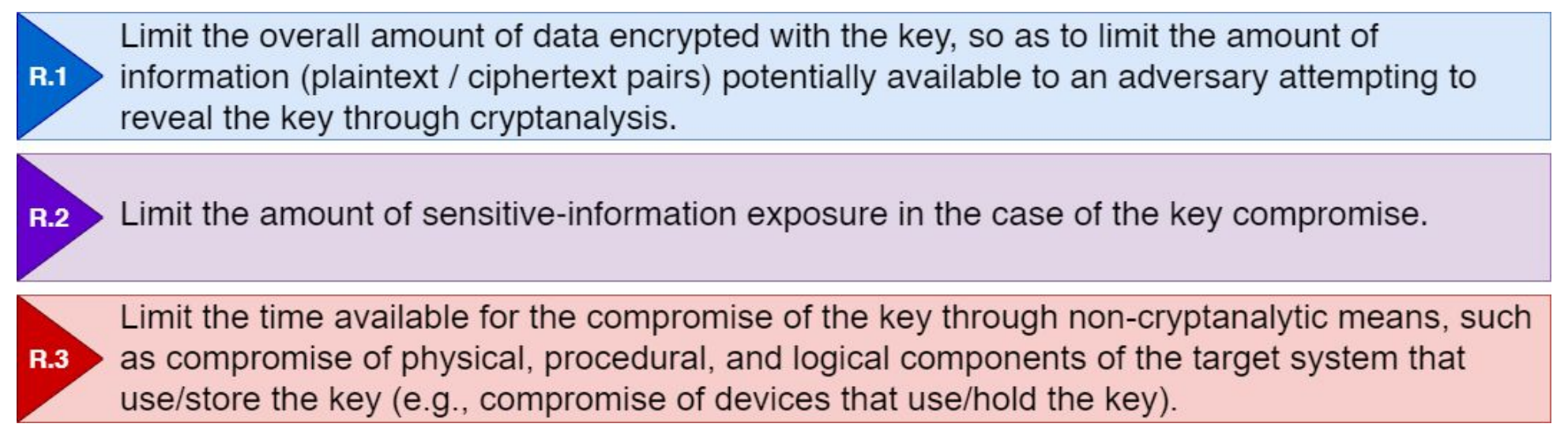
- **NIST SP 800-82r3** and **NIST SP 800-57** standards suggest periodic key rotation - limiting the respective **cryptoperiod**

CRYPTOPERIODS

NIST SP 800.57 | A cryptoperiod is the time span during which a specific key is authorized for use.

How do you calculate the cryptoperiod?

There are **no** explicit guidelines on how to determine the cryptoperiod in a real-world system.



R.1 Limit the overall amount of data encrypted with the key, so as to limit the amount of information (plaintext / ciphertext pairs) potentially available to an adversary attempting to reveal the key through cryptanalysis.

R.2 Limit the amount of sensitive-information exposure in the case of the key compromise.

R.3 Limit the time available for the compromise of the key through non-cryptanalytic means, such as compromise of physical, procedural, and logical components of the target system that use/store the key (e.g., compromise of devices that use/hold the key).

Some existing literature tackles the problem in **general terms**:

- Cryptoperiods should range from **days to weeks for large data** volumes and **up to two years for smaller volumes**
- The *actual determination* of cryptoperiods should involve the **risk** and **consequences** of *key exposure*

While no *tangible* calculation is given, one thing is clear. **Risk** is the underlying factor which affects a cryptoperiod.

- In **higher risk** environments, the cryptoperiod should become **shorter** and **vice versa**

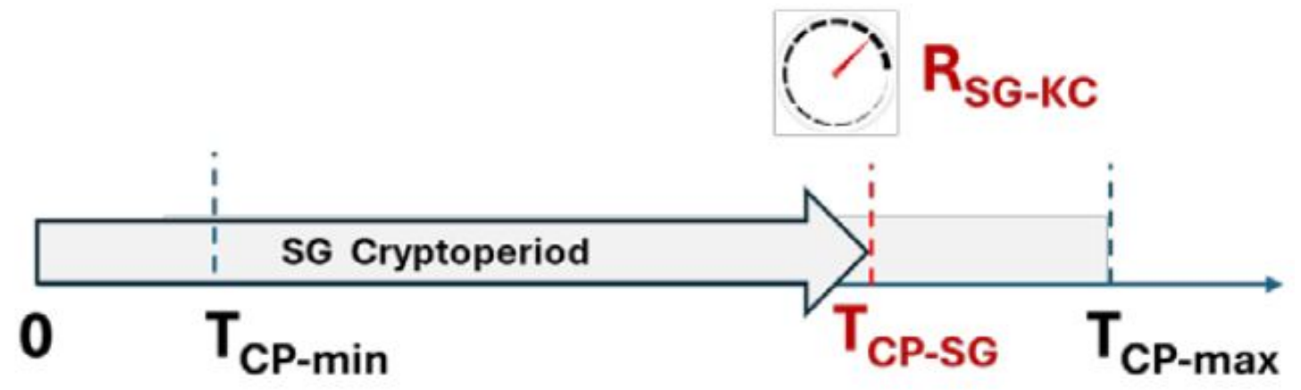
We have developed a **framework** and a respective **software tool** for **A**utomated **R**isk-Based **C**ryptoperiod **C**alculation.

AUTOMATED RISK-BASED CRYPTOPERIOD CALCULATION (ARC-C)

The **framework** and **software tool** are designed to determine the **optimal cryptoperiod** for each scenario, ensuring a balance between the specific **security risks** and the **operational constraints** of the system.

FRAMEWORK

The first step is to establish the **upper** and **lower** *'operational feasibility'* limit on the cryptoperiod.



The **cryptoperiod** *($T_{CP\text{-}SG}$)* is determined by the estimated risk of the environment, following an **exponential relationship**.

The risk *($R_{SG\text{-}KC}$)* can be calculated using: $R_{SG\text{-}KC} = P_{SG\text{-}KC\text{-}succ} \times AI_{SG\text{-}KC}$

- $P_{SG\text{-}KC\text{-}succ}$ is the **probability** that an adversary compromises a key
- $AI_{SG\text{-}KC}$ is the **adverse impact** incurred on the organisation if an attack were to succeed

$P_{SG\text{-}KC\text{-}succ}$ calculation:

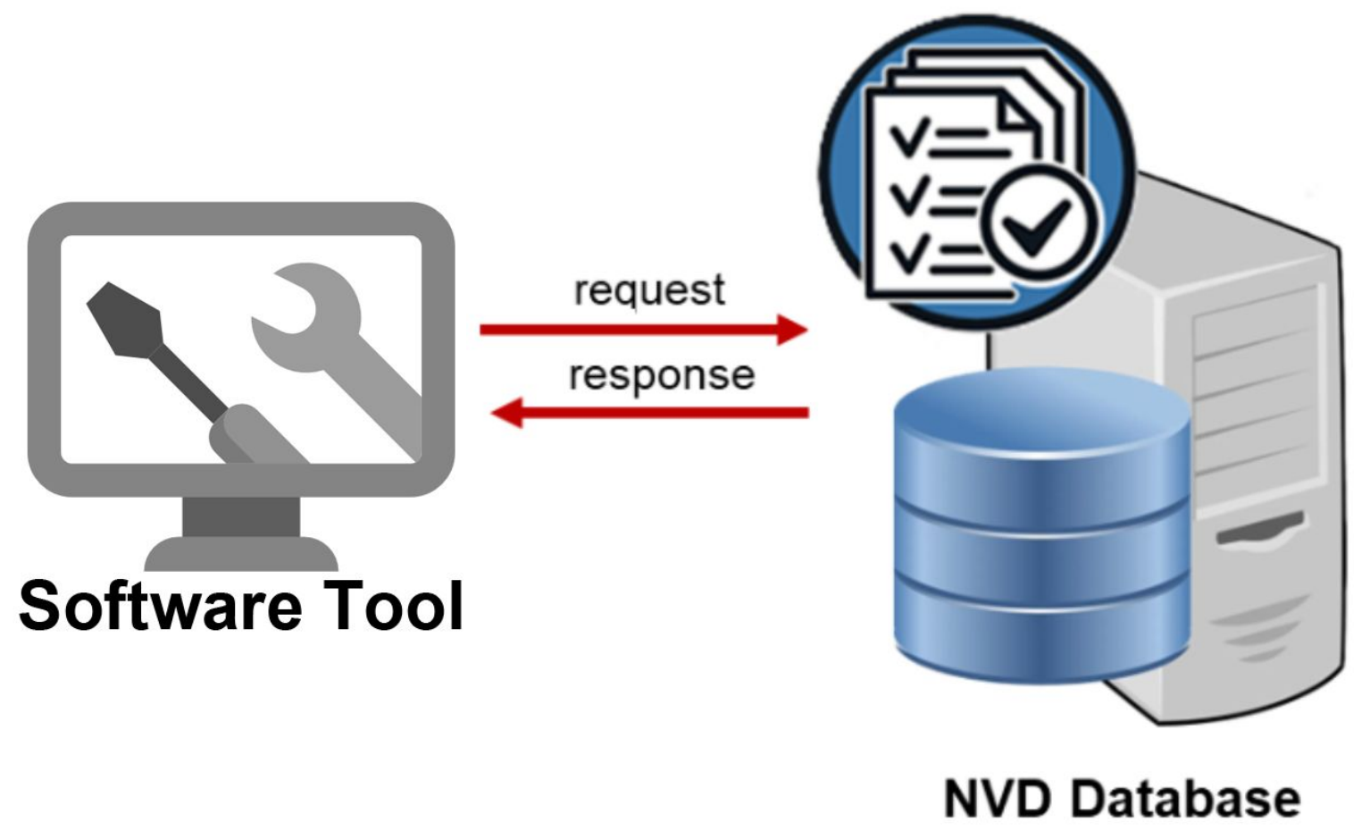
Our **attack tree** model implies two approaches to compromise an encryption key:

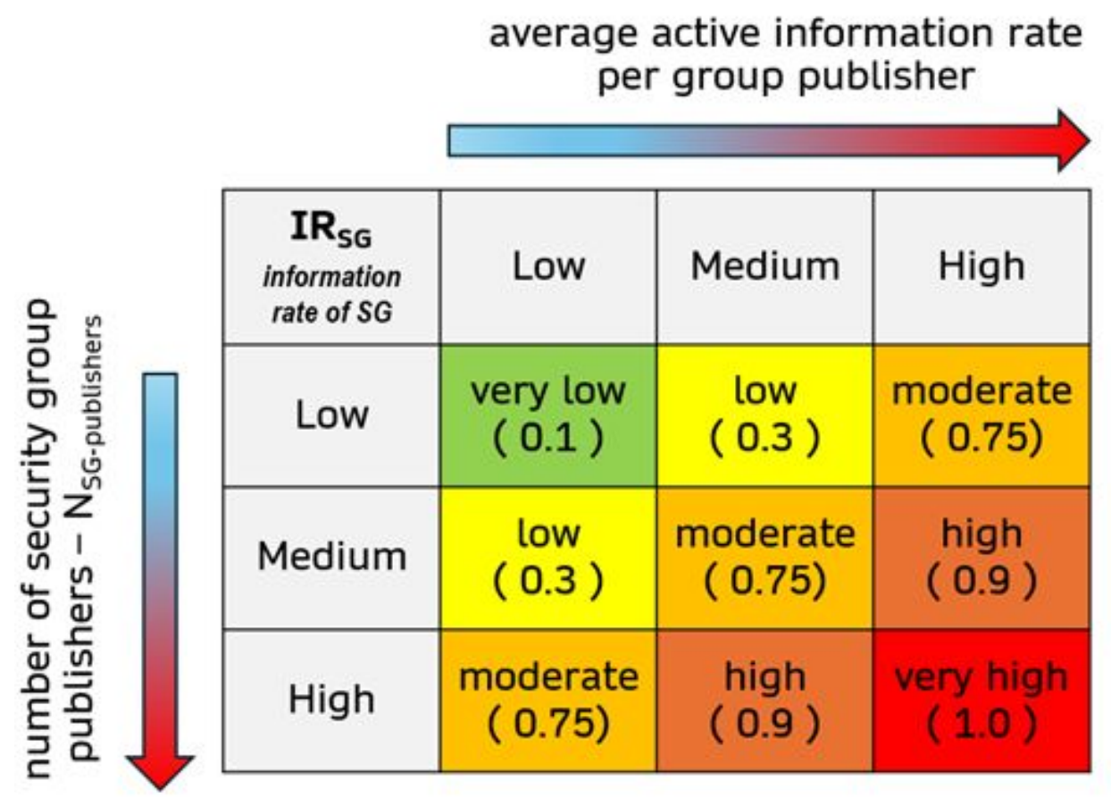
- via **software** vulnerabilities
- via **procedural** vulnerabilities

$AI_{SG\text{-}KC}$ calculation:

The **adverse impact** consists of:

- **Information Rate** or "how much data is being sent"
- **Functional Importance** or "how important is the data"




average active information rate per group publisher
number of security group publishers – $N_{SG\text{-}publishers}$

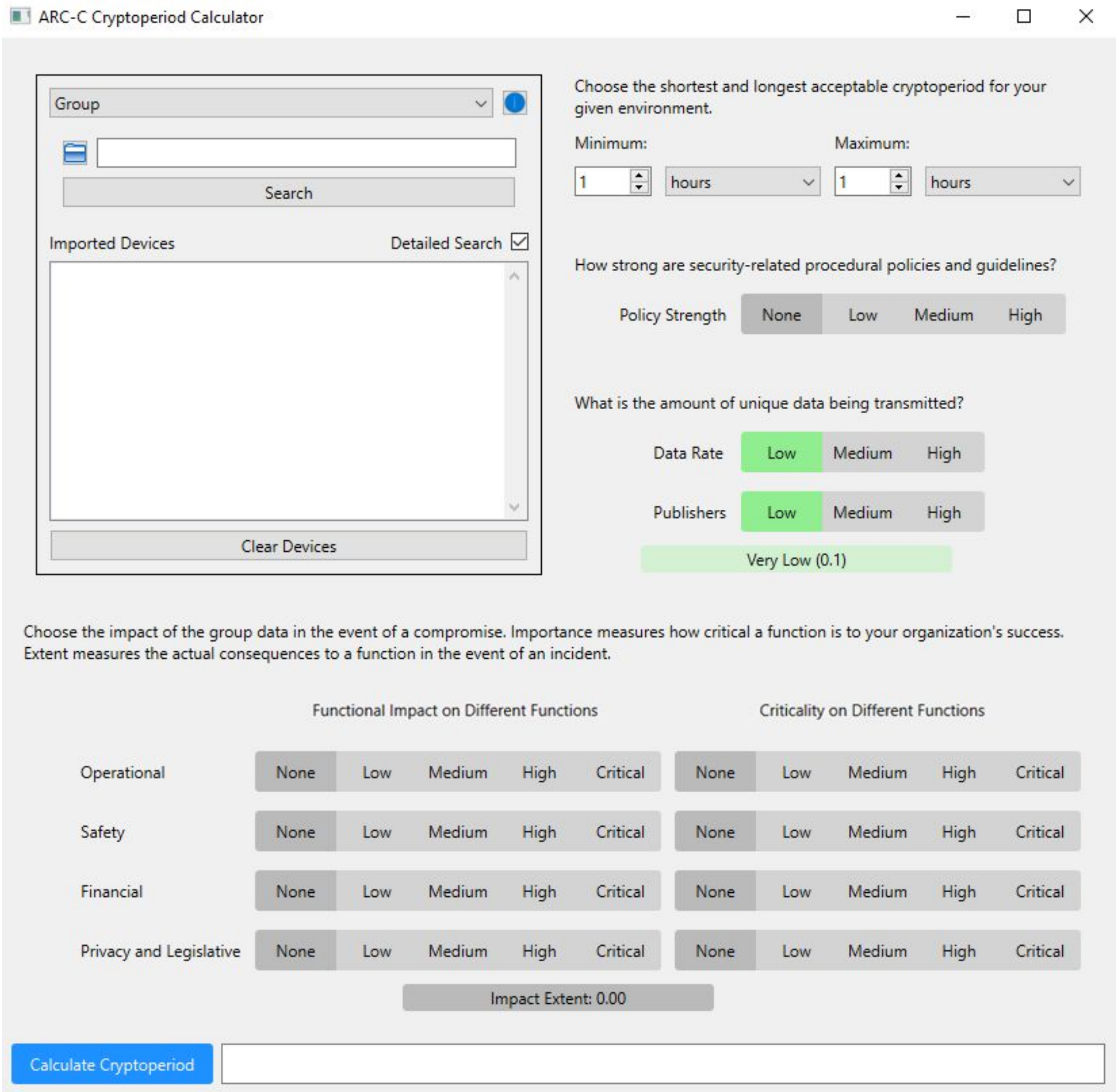
IR_{SG} *information rate of SG*	Low	Medium	High
Low	very low (0.1)	low (0.3)	moderate (0.75)
Medium	low (0.3)	moderate (0.75)	high (0.9)
High	moderate (0.75)	high (0.9)	very high (1.0)

- The **software probability** utilises the **NVD API** for its calculation
- The remaining variables are **set by the operator**, who selects **qualitative values** via our user interface

SOFTWARE TOOL

A **software tool** has been developed, combining the logic with a **user-friendly interface.**

- Parameters are grouped to ensure **logical flow**, with each aspect of calculation **self-contained**



ARC-C Cryptoperiod Calculator

Group

Search

Imported Devices — Detailed Search ☑

Clear Devices

Choose the shortest and longest acceptable cryptoperiod for your given environment.

Minimum: 1 hours — Maximum: 1 hours

How strong are security-related procedural policies and guidelines?

Policy Strength: None | Low | Medium | High

What is the amount of unique data being transmitted?

Data Rate: Low | Medium | High

Publishers: Low | Medium | High

Very Low (0.1)

Choose the impact of the group data in the event of a compromise. Importance measures how critical a function is to your organization's success. Extent measures the actual consequences to a function in the event of an incident.

	Functional Impact on Different Functions					Criticality on Different Functions				
Operational	None	Low	Medium	High	Critical	None	Low	Medium	High	Critical
Safety	None	Low	Medium	High	Critical	None	Low	Medium	High	Critical
Financial	None	Low	Medium	High	Critical	None	Low	Medium	High	Critical
Privacy and Legislative	None	Low	Medium	High	Critical	None	Low	Medium	High	Critical

Impact Extent: 0.00

Calculate Cryptoperiod

To test the **validity** of the framework, we created two hypothetical but highly plausible ICSs:

- **Water Treatment Facility**
- **Energy Storage System** (ESS)

For a more **in-depth** look and a deep dive into the **experimental results**, *scan the QR code.*