PRIVACY-FRIENDLY LOGGING FOR SECURITY

Joeri de Ruiter

How it started

- Ransomware attack at Dutch university
- SURFcert identified domain names that could indicate infection
- Domain names shared with other organisations
- SURF asked which systems looked these up
- At SURF this was not logged
- **How can we see what systems requested a specific domain name?**

Why do we want this?

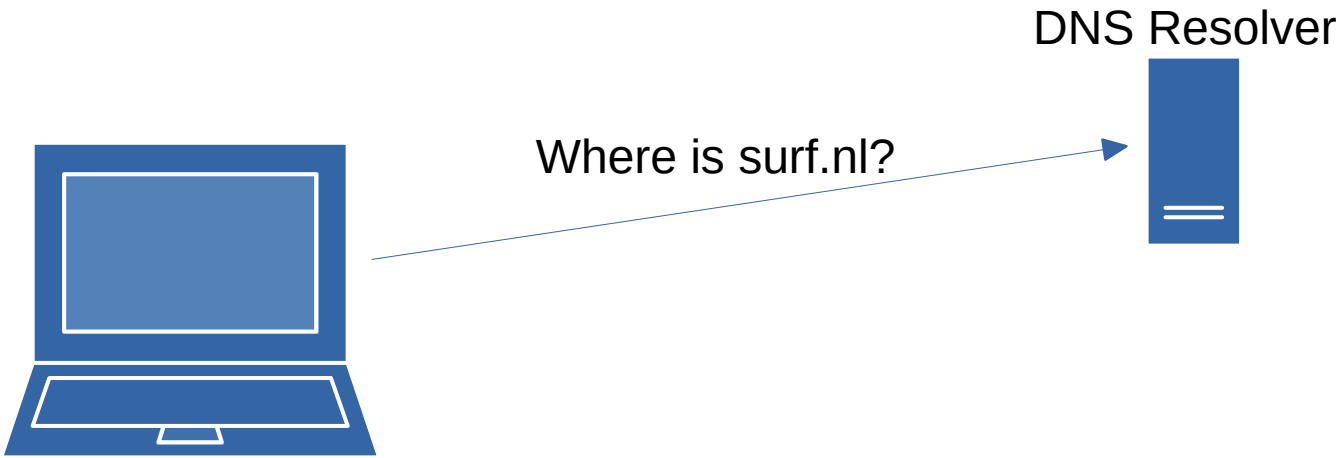
- Early detection of infected systems
  - Malware often requests specific domain names
  - For example to connect with command & control servers
- Real-time alerting
  - Known malicious domain names
- Look back
  - For newly identified malicious domain names

DNS intro

- Phonebook of the Internet
- Translation from domain names to IP addresses
- For example: surf.nl → 2001:610:508:108:192:87:108:15

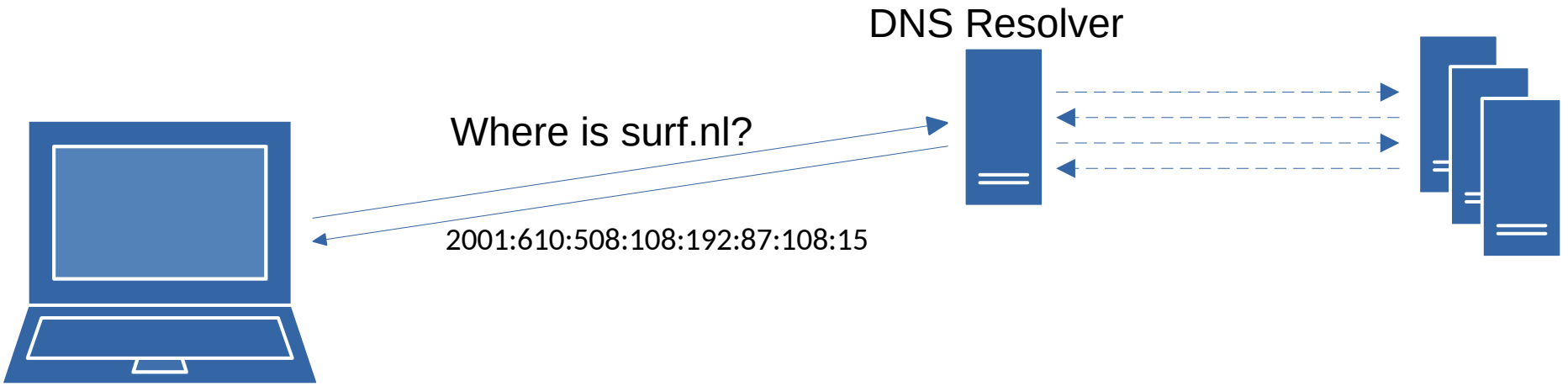
DNS intro

- Phonebook of the Internet
- Translation from domain names to IP addresses
- For example: surf.nl → 2001:610:508:108:192:87:108:15



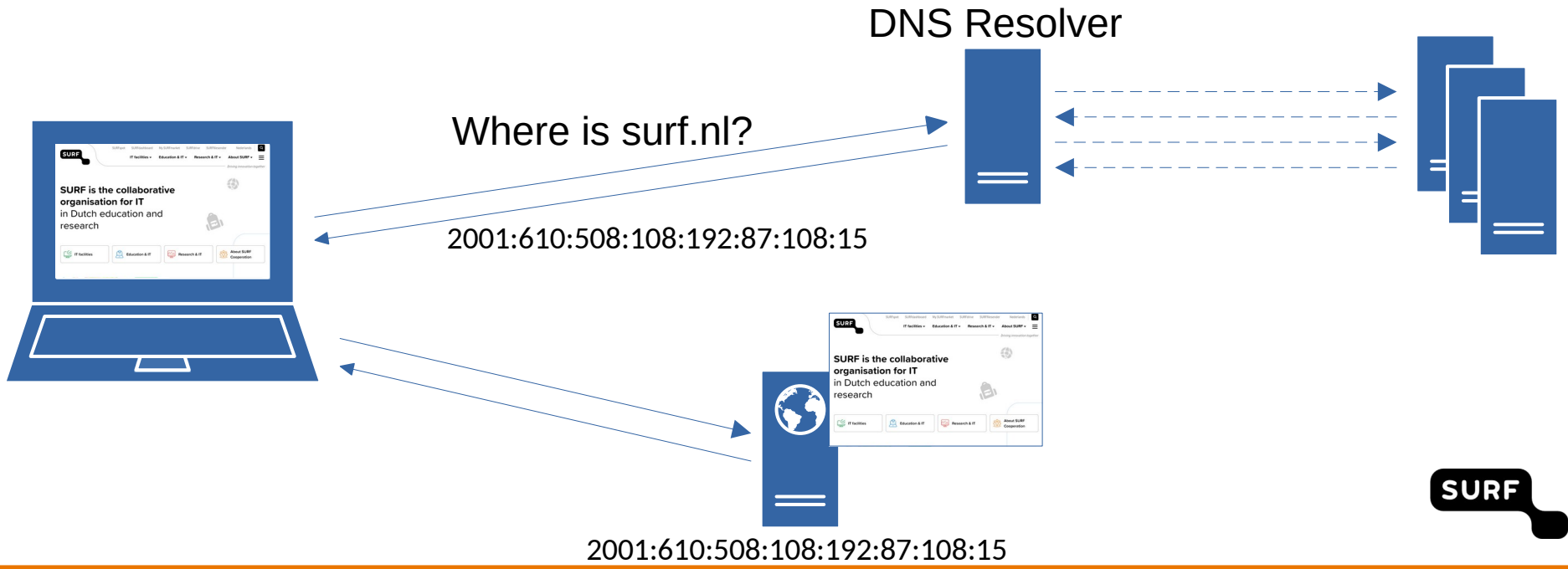
DNS intro

- Phonebook of the Internet
- Translation from domain names to IP addresses
- For example: surf.nl → 2001:610:508:108:192:87:108:15



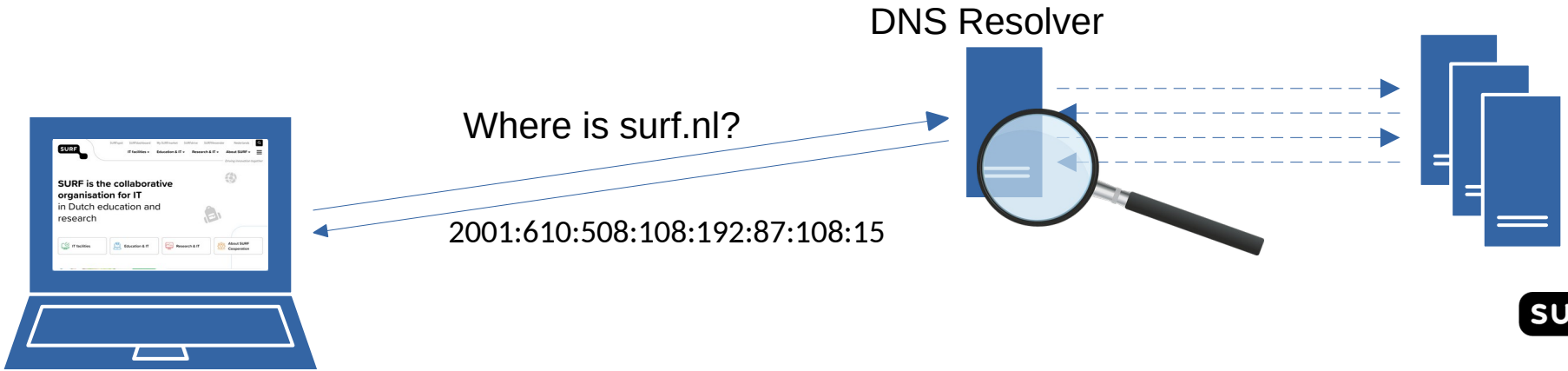
DNS intro

- Phonebook of the Internet
- Translation from domain names to IP addresses
- For example: surf.nl → 2001:610:508:108:192:87:108:15



Who requested a particular domain name?

- Logging queries at the DNS resolver
- Very useful to find possibly compromised systems
  - Domain names not always immediately known
- However: very privacy sensitive
  - It reveals the users' browsing behaviour
- **How can we log and monitor DNS queries in a privacy-friendly manner?**

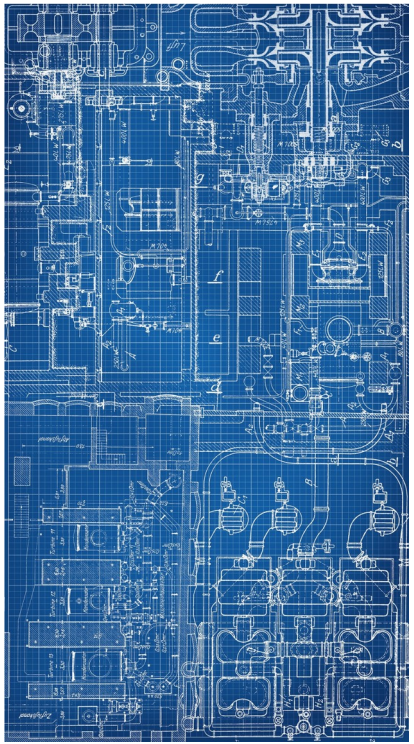


What data are we interested in?

- Requested domain name
- Time of request
- Information to find the user, such as the IP address
- Easy solution: put it all in a database
  - Not very privacy-friendly
  - Can we do better?

What do we need?

- Privacy-by-design
- The only question we are interested in is:
  - Which users requested a particular (malicious) domain name?
- We are not interested in knowing which domain names a particular user requested
- Both questions require the same data: domain name and user info
- **Can we design a system in which we can answer the first question, but not the second?**

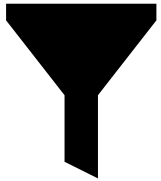


Intermezzo: cryptographic hash functions

- Transform some text into a random looking string of characters
  - Always the same string for the same input
  - Outputs of the same length
  - One-way: you cannot easily get the original input given just an output

Intermezzo: cryptographic hash functions

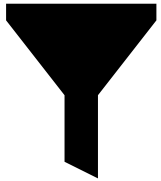
surf.nl



51EF8FD9D7B9996C89DB2F0AD3F91C50

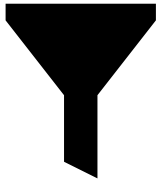
Intermezzo: cryptographic hash functions

surf.nl



51EF8FD9D7B9996C89DB2F0AD3F91C50

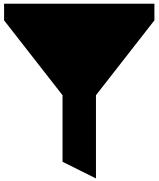
surfdomeinen.nl



CEAC9BF76D801DAB8F5C9889472DF84A

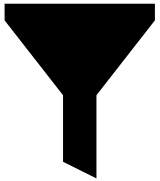
Intermezzo: cryptographic hash functions

surf.nl



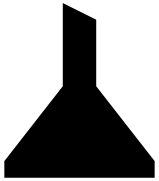
51EF8FD9D7B9996C89DB2F0AD3F91C50

surfdomeinen.nl



CEAC9BF76D801DAB8F5C9889472DF84A

F20404BDD1003CDF110C929F8C238C98



?

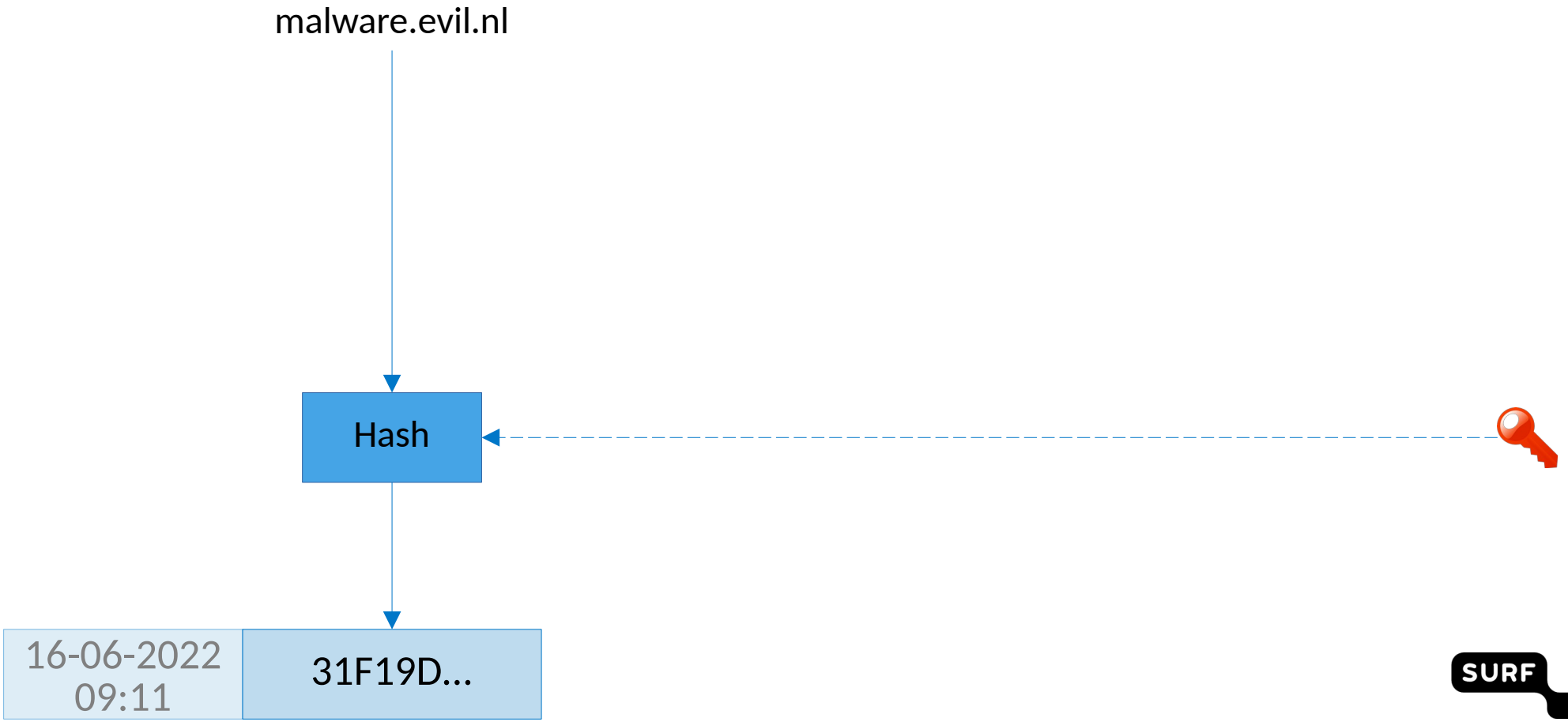
Privacy-friendly storage

- Store the hash of the requested domain name
  - Possible to search for specific domain names
  - Not possible to see which domain names a particular user requested
  - Additional layer of security by protecting the hashes with a key
- Information about the user is encrypted
  - Key based on the domain name
  - Only decrypted if you know for which domain name it was stored
- Domain name itself not stored
  - Only derived information

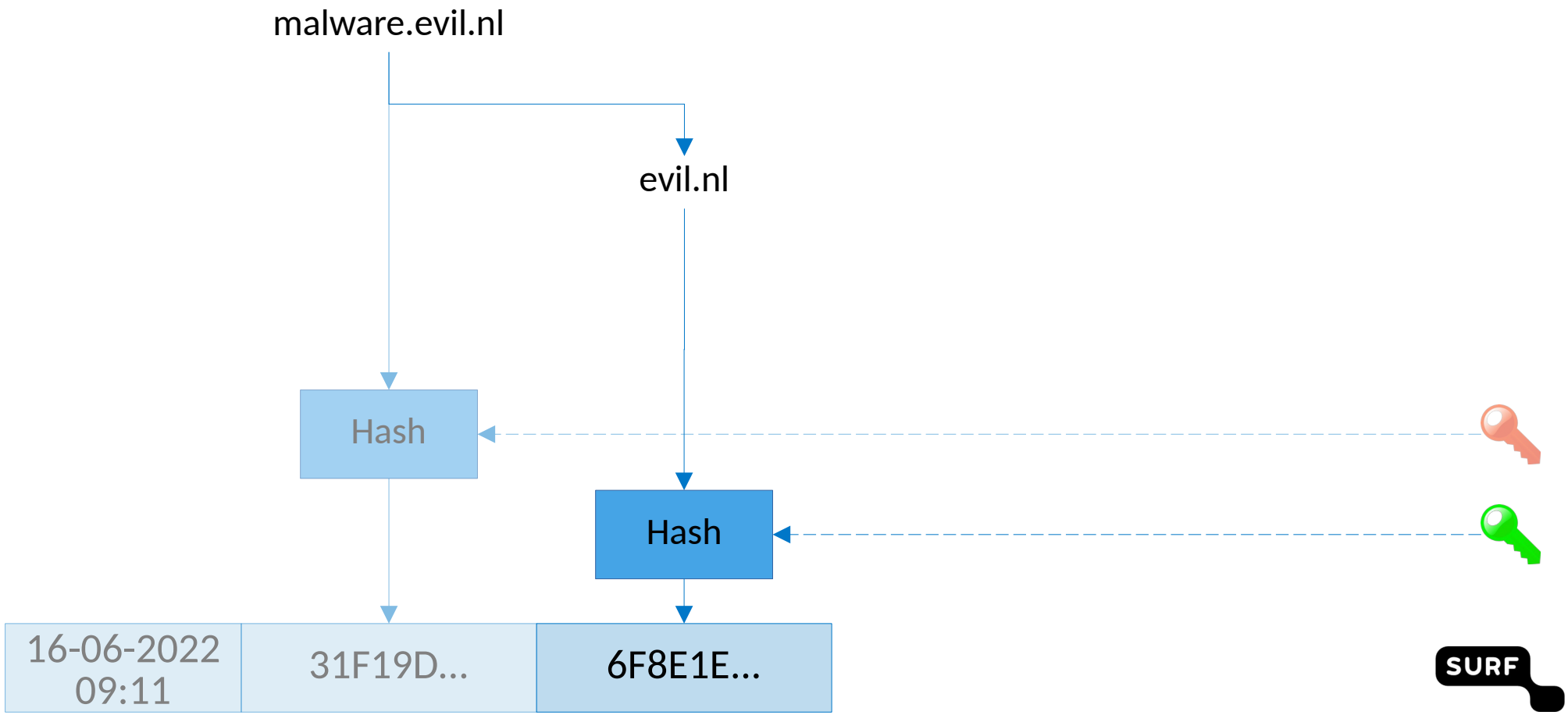
192.0.2.123 looks up malware.evil.nl

16-06-2022
09:11

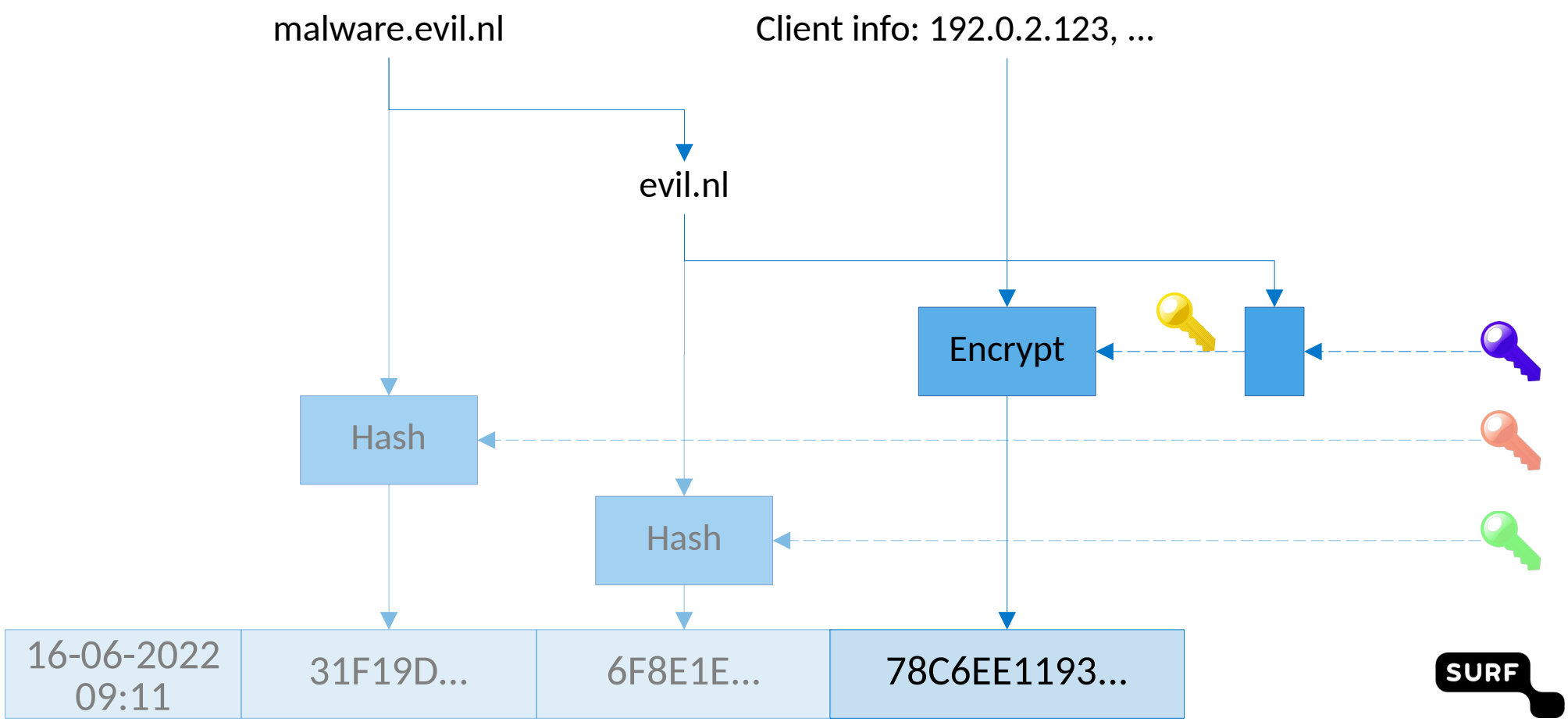
192.0.2.123 looks up malware.evil.nl



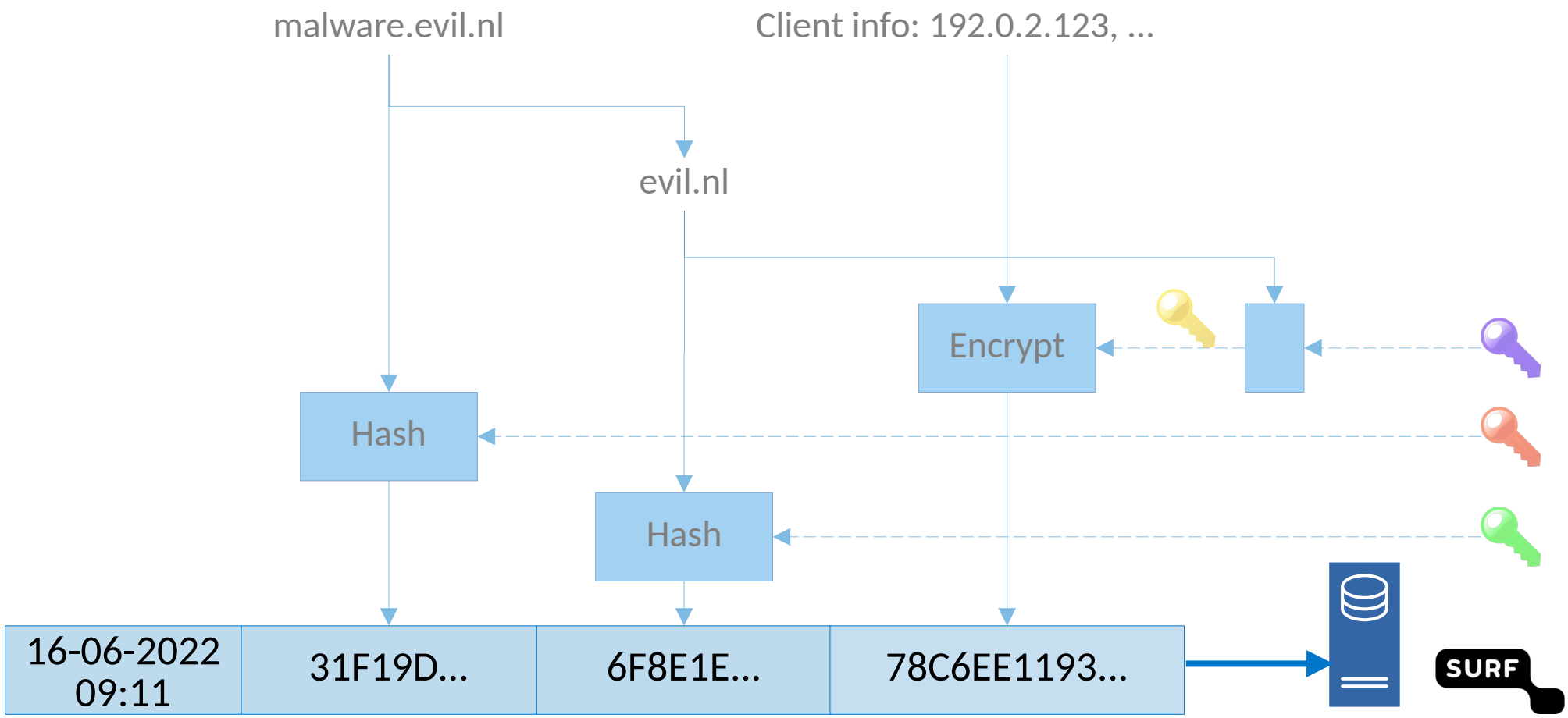
192.0.2.123 looks up malware.evil.nl



192.0.2.123 looks up malware.evil.nl



192.0.2.123 looks up malware.evil.nl

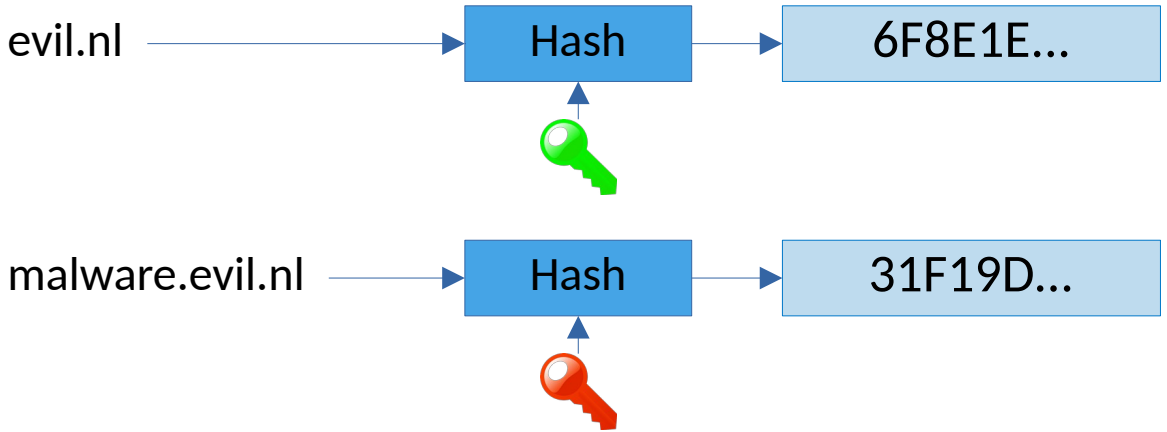


Who looked up malware.evil.nl?

Time	Domain		User info
16-06-2022 09:05	A623BE…	E47422…	DF55912A1B…
16-06-2022 09:10	38FE31…	12A8B3…	56ED4CB784…
16-06-2022 09:11	31F19D…	6F8E1E…	78C6EE1193…
16-06-2022 09:14	6789DE…	BF5136…	C45DE823FF…

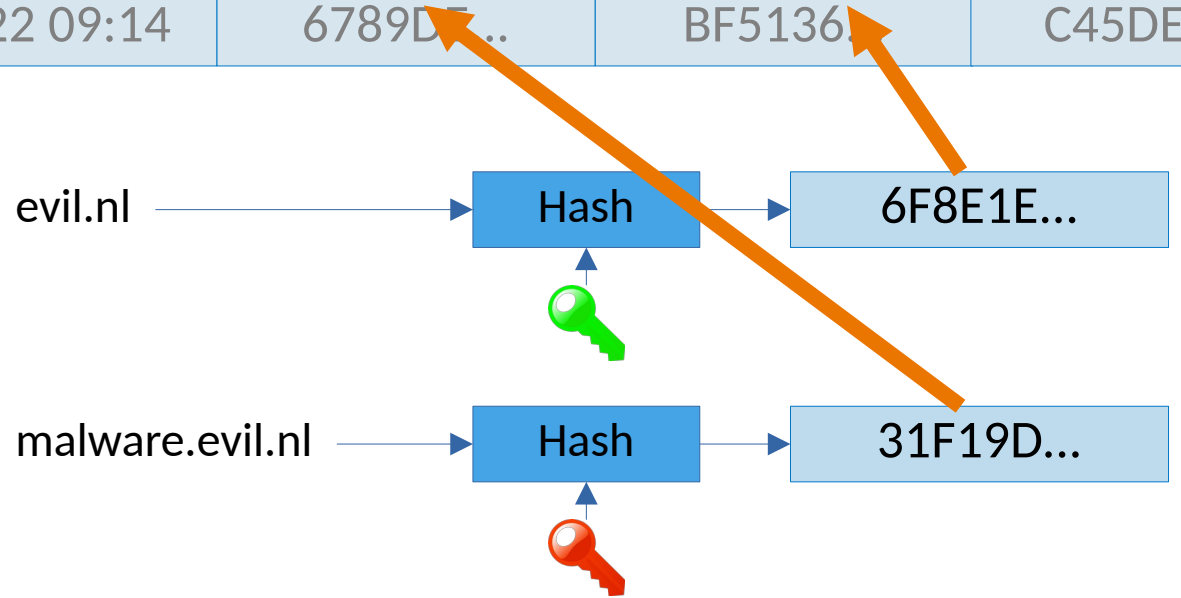
Who looked up malware.evil.nl?

Time	Domain		User info
16-06-2022 09:05	A623BE…	E47422…	DF55912A1B…
16-06-2022 09:10	38FE31…	12A8B3…	56ED4CB784…
16-06-2022 09:11	31F19D…	6F8E1E…	78C6EE1193…
16-06-2022 09:14	6789DE…	BF5136…	C45DE823FF…



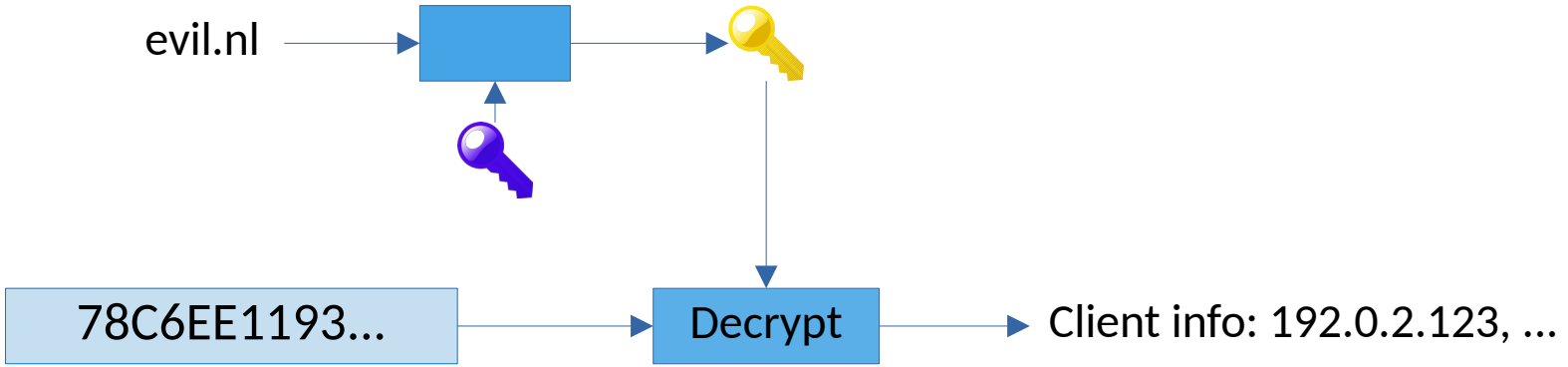
Who looked up malware.evil.nl?

Time	Domain		User info
16-06-2022 09:05	A623BE…	E47422…	DF55912A1B…
16-06-2022 09:10	38FE31…	12A8B3…	56ED4CB784…
16-06-2022 09:11	31F19D…	6F8E1E…	78C6EE1193…
16-06-2022 09:14	6789D…	BF5136…	C45DE823FF…



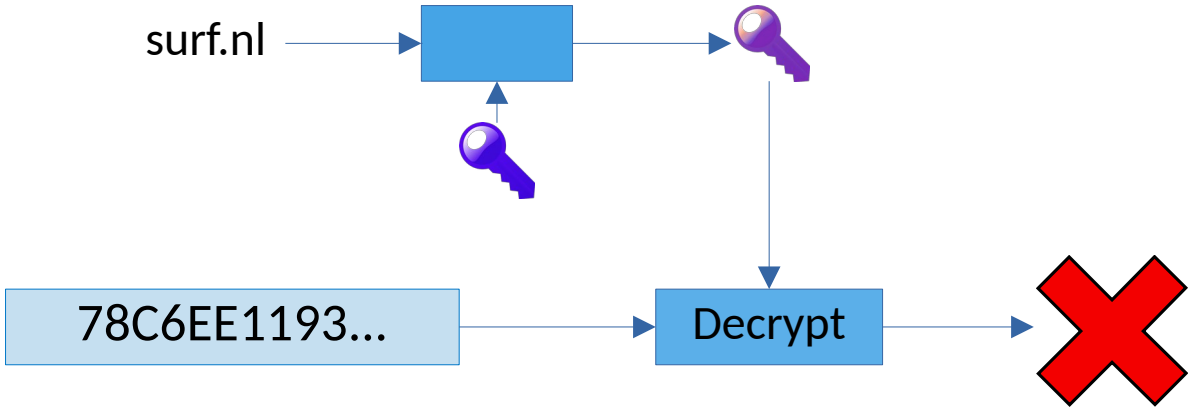
Who looked up malware.evil.nl?

Time	Domain		User info
16-06-2022 09:05	A623BE…	E47422…	DF55912A1B…
16-06-2022 09:10	38FE31…	12A8B3…	56ED4CB784…
16-06-2022 09:11	31F19D…	6F8E1E…	78C6EE1193…
16-06-2022 09:14	6789DE…	BF5136…	C45DE823FF…



Who looked up malware.evil.nl?

Time	Domain		User info
16-06-2022 09:05	A623BE…	E47422…	DF55912A1B…
16-06-2022 09:10	38FE31…	12A8B3…	56ED4CB784…
16-06-2022 09:11	31F19D…	6F8E1E…	78C6EE1193…
16-06-2022 09:14	6789DE…	BF5136…	C45DE823FF…



Privacy-friendly storage

- Who requested malware.evil.nl?
  - Need access to the keys

- Which domain names did 192.0.2.123 request?
  - User information is encrypted using the requested domain name

Additional safeguards

- Limited access to system
- Audit log
  - Who searched for what and when in the database?
  - Regular checks
- Data stored for limited time
  - No longer than 90 days

Some technical details

- DNS queries exported using DNSTAP
- Hashes using HMAC-SHA256
- Encryption using ChaCha20Poly1305, an authenticated encryption scheme
- Data per row: 139 bytes
  - ~4 TB for 90 days w/o overhead
- Stored in PostgreSQL database
  - Table partitioned per day
  - Index on partitions
- Goal to release as open source

Eat your own dog food

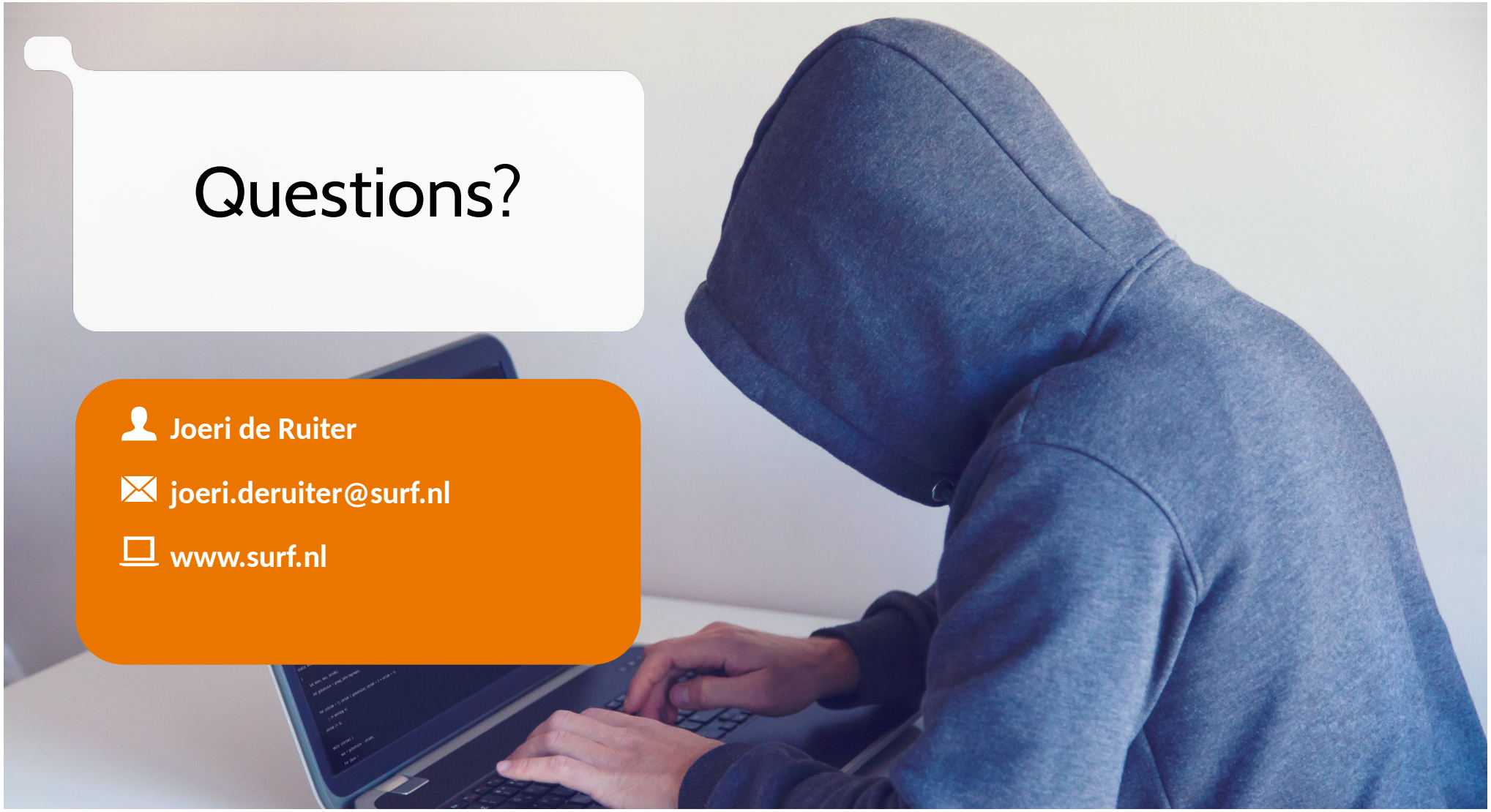
- Initial discussions with data protection officer positive
- Pilot for the internal SURF network
  - Real-time checks against known malicious domain names
- Internal presentation and blog post
- Regularly alerts for low level threats
- One alert with high threat level
  - Turned out to be a guest
- Data Protection Impact Assessment (DPIA) in progress

Performance

- DNSTAP non-blocking
  - Minimal impact on resolver
- Only symmetric encryption
- Simulations
  - ~5 billion queries for 1 million unique domains over 20 days, ~1TB data
    - ~5 seconds with index on all tables
    - ~50 seconds on single table without index
- Room for optimisations

Takeaways

- Think in advance which questions you need to answer
  - What data do you really need?
  - How can you store the data to limit possible abuse?
- Be transparent to users
- Security and privacy can go hand in hand


Questions?
Joeri de Ruiter
joeri.deruiter@surf.nl
www.surf.nl