# Security of Symmetric Encryption in the Presence of Ciphertext Fragmentation

Alexandra Boldyreva, **Jean Paul Degabriele**, Kenny Paterson, and Martijn Stam

EUROCRYPT - 19th April 2012

# Outline of this Talk

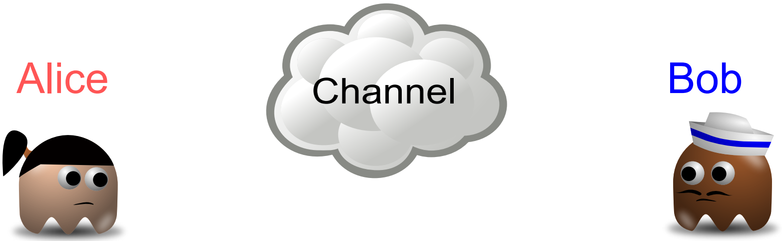

1 Ciphertext Fragmentation and Related Problems

2 Formalizing Fragmentation

3 Security Notions

4 Constructions and Comparison

# Ciphertext Fragmentation



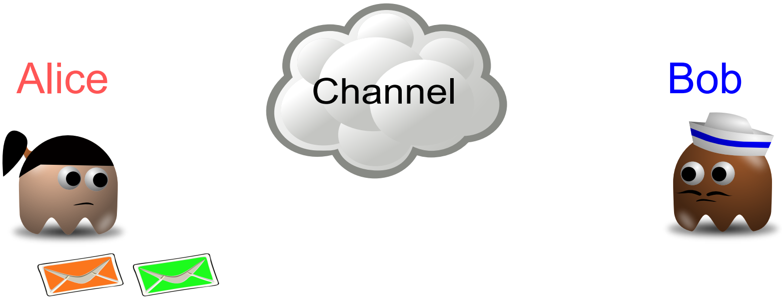

Under *normal operation* the channel delivers ciphertexts in a fragmented fashion, where:

a) The fragmentation pattern is arbitrary.

b) But the order of the fragments is preserved.

# Ciphertext Fragmentation



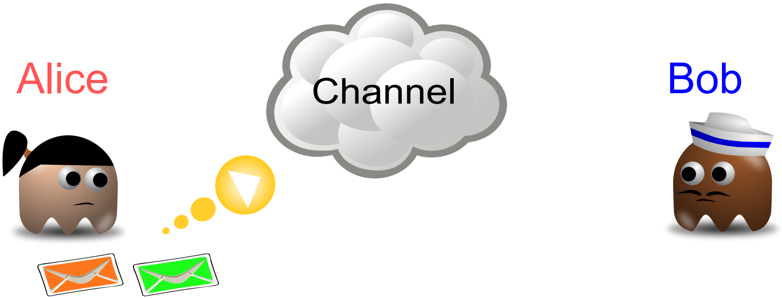



Under *normal operation* the channel delivers ciphertexts in a fragmented fashion, where:

a) The fragmentation pattern is arbitrary.

b) But the order of the fragments is preserved.

# Ciphertext Fragmentation




Under *normal operation* the channel delivers ciphertexts in a fragmented fashion, where:

a) The fragmentation pattern is arbitrary.

b) But the order of the fragments is preserved.

# Ciphertext Fragmentation


Under *normal operation* the channel delivers ciphertexts in a fragmented fashion, where:

a) The fragmentation pattern is arbitrary.

b) But the order of the fragments is preserved.

# Ciphertext Fragmentation


Under *normal operation* the channel delivers ciphertexts in a fragmented fashion, where:

a) The fragmentation pattern is arbitrary.

b) But the order of the fragments is preserved.

# Why Should We Care?

- This setting emerges in practice, where encryption schemes have to operate under such conditions.
- One such instance is that of **secure network protocols**.
- However this is NOT captured by the security models currently used in cryptographic theory!
- Ciphertext fragmentation has given rise to a class of attacks that proved to be **fatal** in certain cases.
- This has left a **gap** between cryptographic theory and practice.

# Ciphertext-Fragmentation Attacks

SSH:

- A proof of security (IND-sfCCA) for SSH was given in **[BKN 04]**.
- Yet **[APW 09]** presented plaintext-recovery attacks against SSH.

IPsec in MAC-then-encrypt (CBC):

- **[Kra 01]** proves that MAC-then-encrypt with CBC encryption is secure (secure channel [CK 01]).
- **[MT 10]** show that MAC-then-encode-then-encrypt (injective / CBC) is secure (secure channel [Mau 11]).
- **[DP 10]** present ciphertext-fragmentation attacks against such IPsec configurations.

# Ciphertext-Fragmentation Attacks

SSH:

- A proof of security (IND-sfCCA) for SSH was given in **[BKN 04]**.
- Yet **[APW 09]** presented plaintext-recovery attacks against SSH.

IPsec in MAC-then-encrypt (CBC):

- **[Kra 01]** proves that MAC-then-encrypt with CBC encryption is secure (secure channel [CK 01]).
- **[MT 10]** show that MAC-then-encode-then-encrypt (injective / CBC) is secure (secure channel [Mau 11]).
- **[DP 10]** present ciphertext-fragmentation attacks against such IPsec configurations.

# Ciphertext-Fragmentation Attacks



SSH:

- A proof of security (IND-sfCCA) for SSH was given in **[BKN 04]**.
- Yet **[APW 09]** presented plaintext-recovery attacks against SSH.

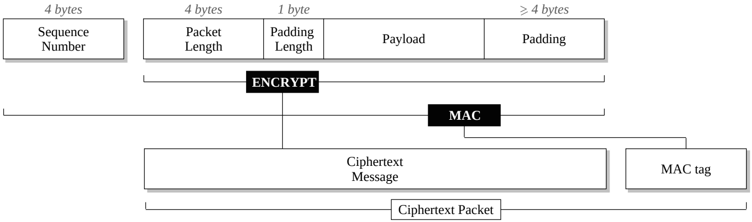
IPsec in MAC-then-encrypt (CBC):

- **[Kra 01]** proves that MAC-then-encrypt with CBC encryption is secure (secure channel [CK 01]).
- **[MT 10]** show that MAC-then-encode-then-encrypt (injective / CBC) is secure (secure channel [Mau 11]).
- **[DP 10]** present ciphertext-fragmentation attacks against such IPsec configurations.

# The SSH Attack (Main Idea)

Royal Holloway University of London Information Security Group

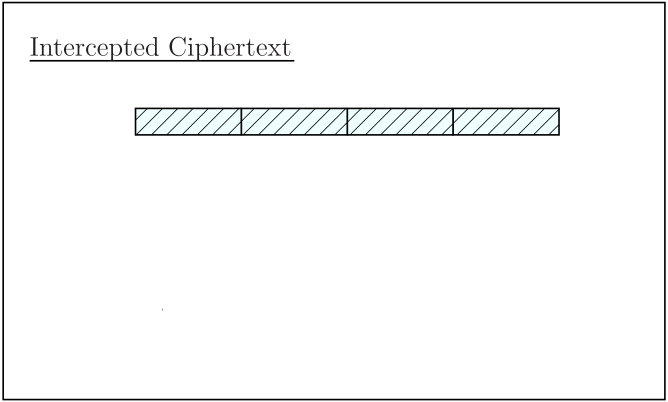
- SSH encrypts messages in the following format:




- SSH commonly uses CBC mode for encryption.

# The SSH Attack (Main Idea)

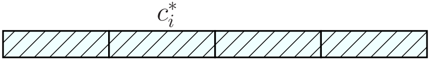
Royal Holloway University of London Information Security Group



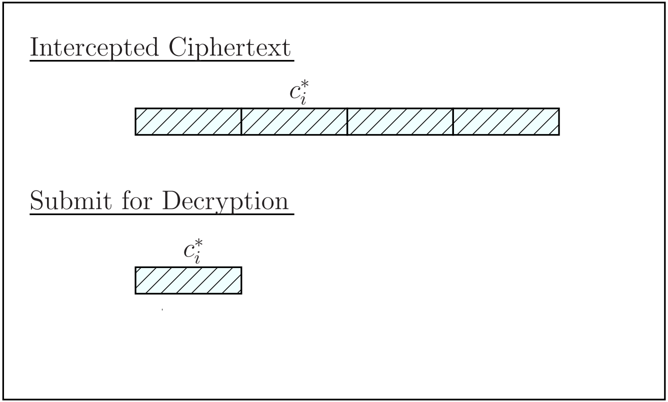
# The SSH Attack (Main Idea)

Royal Holloway University of London Information Security Group

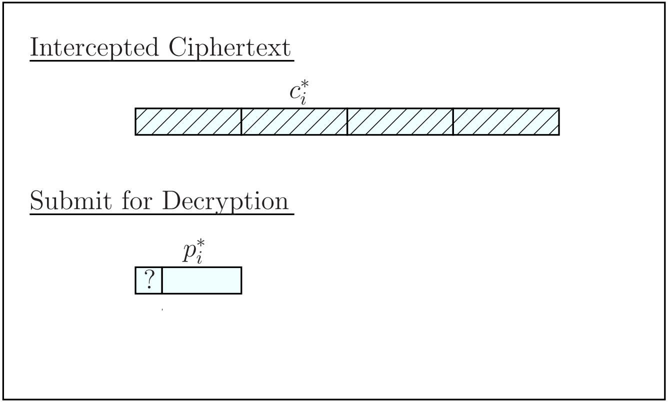
Intercepted Ciphertext



# The SSH Attack (Main Idea)

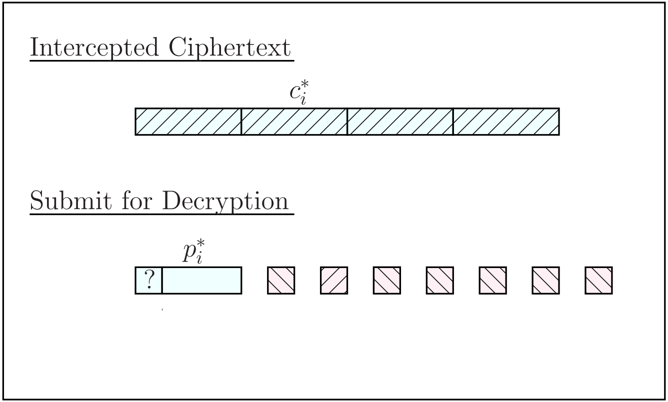


# The SSH Attack (Main Idea)



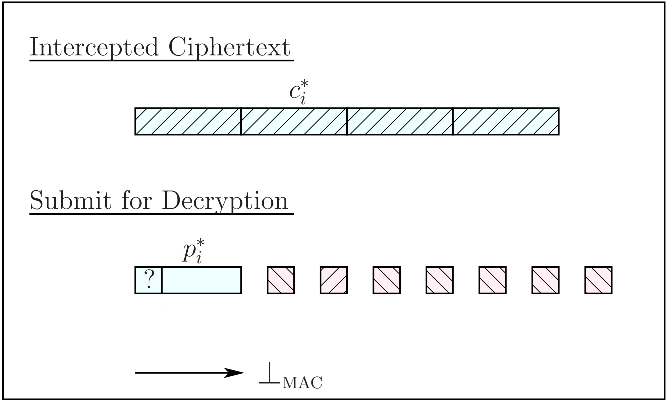
# The SSH Attack (Main Idea)

Royal Holloway University of London Information Security Group



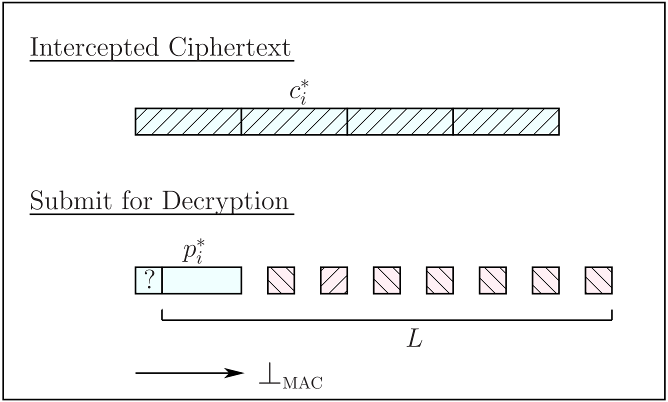
# The SSH Attack (Main Idea)



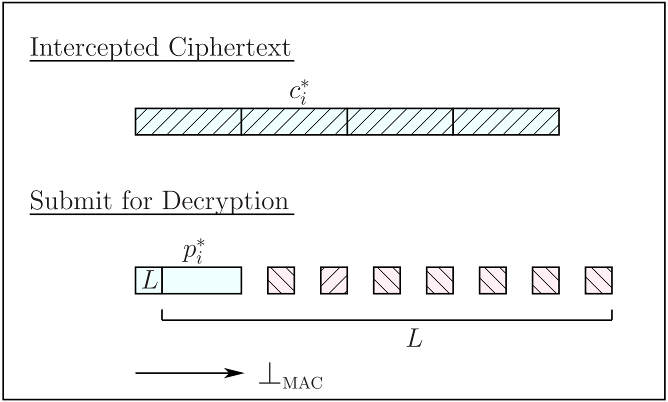


# The SSH Attack (Main Idea)

Royal Holloway University of London Information Security Group



# The SSH Attack (Main Idea)



# Related Work

Royal Holloway University of London Information Security Group

- A first step towards analyzing security in the presence of ciphertext fragmentation was made by Paterson and Watson in 2010.
- They show that when CBC mode is replaced with (stateful) **counter mode** SSH is secure.
- However their security notion is closely tied to SSH, and hence it is not generally applicable to other schemes.
- At first glance, ciphertext fragmentation may show some resemblance to **online encryption**. We emphasize that there are some important differences, and the two settings are disjoint.

# Our Contribution

- We define a **syntax** and **security notions** for encryption in the fragmented setting.
- We provide **generic constructions** of fragmented schemes that meet our security notions, from normal “atomic” schemes.
- We formalize other security goals that practical schemes commonly aim to achieve: **boundary-hiding** and robustness against **fragmentation-related DoS attacks**.
- We construct a scheme, **InterMAC**, that meets all three of our security notions.

# Syntax

A **fragmented symmetric encryption scheme** $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ with associated message space $\mathcal{M} = \{0,1\}^*$ and ciphertext space $\mathcal{C} = \{0,1\}^*$, is a triple of algorithms such that:

- $(K, \sigma_0, \tau_0) \leftarrow \mathcal{K}$ where $\sigma_0$ and $\tau_0$ are the respective initial states for encryption and decryption.
- $(c, \sigma_{i+1}) \leftarrow \mathcal{E}_K(m, \sigma_i)$ where $\mathcal{E}_K(\cdot)$ can be probabilistic, stateful, or both ($\sigma = \varepsilon$ for stateless); $m \in \mathcal{M}$, $c \in \mathcal{C}$.
- $(m, \tau_{i+1}) \leftarrow \mathcal{D}_K(f, \tau_i)$ where $\mathcal{D}_K(\cdot)$ is deterministic and stateful; $f \in \{0,1\}^*$ and $m \in (\{0,1\} \cup \mathcal{S}_\perp \cup \{\P\})^*$.

# Syntax



A **fragmented symmetric encryption scheme** $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ with associated message space $\mathcal{M} = \{0,1\}^*$ and ciphertext space $\mathcal{C} = \{0,1\}^*$, is a triple of algorithms such that:

- $(K, \sigma_0, \tau_0) \leftarrow \mathcal{K}$ where $\sigma_0$ and $\tau_0$ are the respective initial states for encryption and decryption.
- $(c, \sigma_{i+1}) \leftarrow \mathcal{E}_K(m, \sigma_i)$ where $\mathcal{E}_K(\cdot)$ can be probabilistic, stateful, or both ($\sigma = \varepsilon$ for stateless); $m \in \mathcal{M}$, $c \in \mathcal{C}$.
- $(m, \tau_{i+1}) \leftarrow \mathcal{D}_K(f, \tau_i)$ where $\mathcal{D}_K(\cdot)$ is deterministic and stateful; $f \in \{0,1\}^*$ and $m \in (\{0,1\} \cup \mathcal{S}_\perp \cup \{\P\})^*$.

# Syntax

A **fragmented symmetric encryption scheme** $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ with associated message space $\mathcal{M} = \{0,1\}^*$ and ciphertext space $\mathcal{C} = \{0,1\}^*$, is a triple of algorithms such that:

- $(K, \sigma_0, \tau_0) \leftarrow \mathcal{K}$ where $\sigma_0$ and $\tau_0$ are the respective initial states for encryption and decryption.
- $(c, \sigma_{i+1}) \leftarrow \mathcal{E}_K(m, \sigma_i)$ where $\mathcal{E}_K(\cdot)$ can be probabilistic, stateful, or both ($\sigma = \varepsilon$ for stateless); $m \in \mathcal{M}$, $c \in \mathcal{C}$.
- $(m, \tau_{i+1}) \leftarrow \mathcal{D}_K(f, \tau_i)$ where $\mathcal{D}_K(\cdot)$ is deterministic and stateful; $f \in \{0,1\}^*$ and $m \in (\{0,1\} \cup \mathcal{S}_\perp \cup \{¶\})^*$.

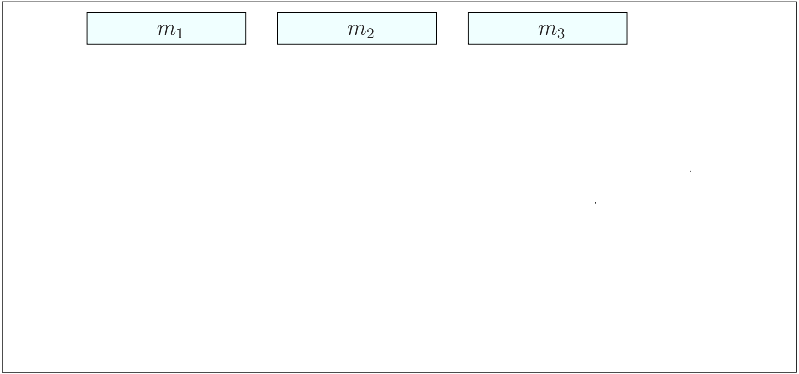
# Syntax

A **fragmented symmetric encryption scheme** $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ with associated message space $\mathcal{M} = \{0,1\}^*$ and ciphertext space $\mathcal{C} = \{0,1\}^*$, is a triple of algorithms such that:

- $(K, \sigma_0, \tau_0) \leftarrow \mathcal{K}$ where $\sigma_0$ and $\tau_0$ are the respective initial states for encryption and decryption.
- $(c, \sigma_{i+1}) \leftarrow \mathcal{E}_K(m, \sigma_i)$ where $\mathcal{E}_K(\cdot)$ can be probabilistic, stateful, or both ($\sigma = \varepsilon$ for stateless); $m \in \mathcal{M}$, $c \in \mathcal{C}$.
- $(m, \tau_{i+1}) \leftarrow \mathcal{D}_K(f, \tau_i)$ where $\mathcal{D}_K(\cdot)$ is deterministic and stateful; $f \in \{0,1\}^*$ and $m \in (\{0,1\} \cup \mathcal{S}_\perp \cup \{\P\})^*$.

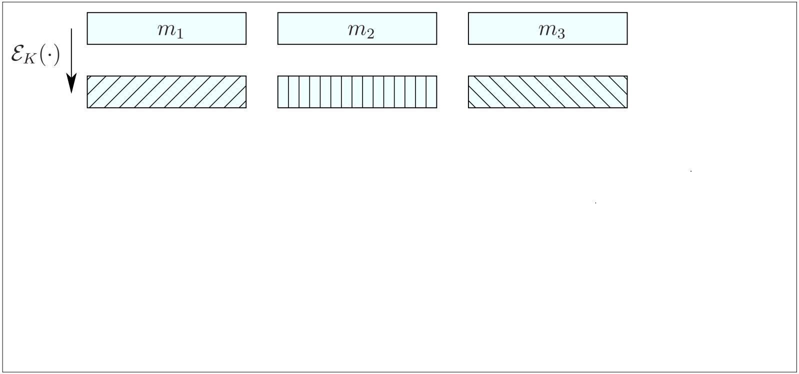
# Correctness Requirement

(explained pictorially)



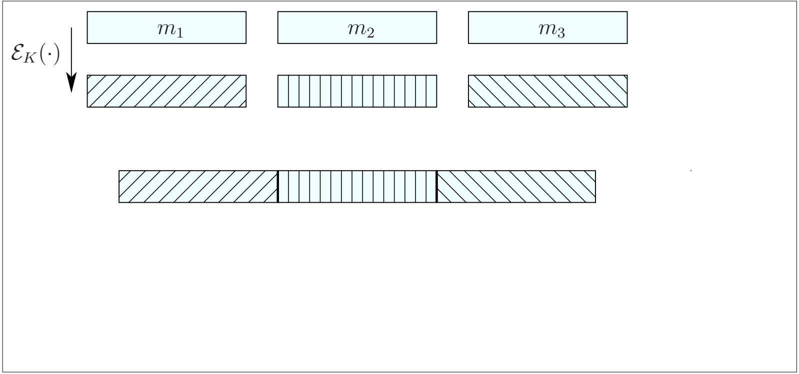

- Then $m_1 \,||\, ¶ \,||\, m_2 \,||\, ¶ \,||\, m_3 \,||\, ¶$ is a prefix of $m'_1 \,||\, m'_2 \,||\, m'_3 \,||\, m'_4 \,||\, m'_5$.

# Correctness Requirement
(explained pictorially)




- Then $m_1 \,||\, \P \,||\, m_2 \,||\, \P \,||\, m_3 \,||\, \P$ is a prefix of $m'_1 \,||\, m'_2 \,||\, m'_3 \,||\, m'_4 \,||\, m'_5$.

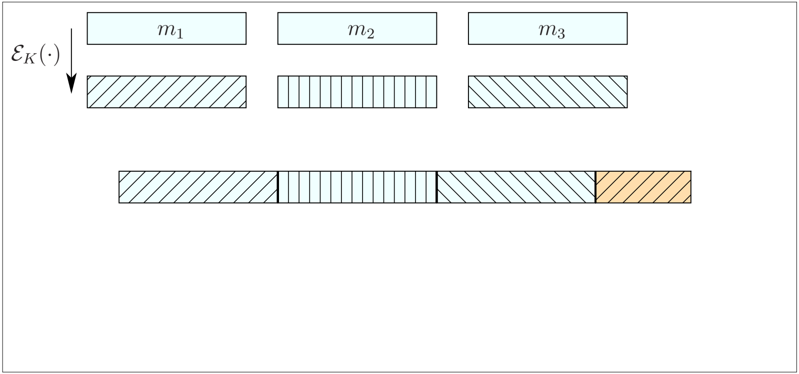
# Correctness Requirement
(explained pictorially)




- Then $m_1 \,||\, ¶ \,||\, m_2 \,||\, ¶ \,||\, m_3 \,||\, ¶$ is a prefix of $m'_1 \,||\, m'_2 \,||\, m'_3 \,||\, m'_4 \,||\, m'_5$.

# Correctness Requirement

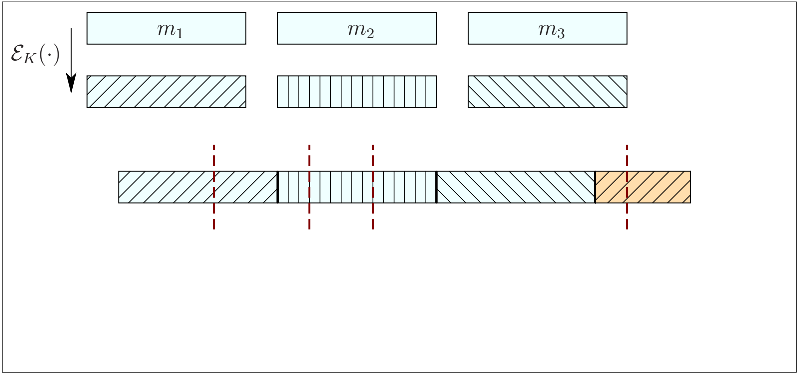
(explained pictorially)




- Then $m_1 \,||\, ¶ \,||\, m_2 \,||\, ¶ \,||\, m_3 \,||\, ¶$ is a prefix of $m'_1 \,||\, m'_2 \,||\, m'_3 \,||\, m'_4 \,||\, m'_5$.

# Correctness Requirement

(explained pictorially)

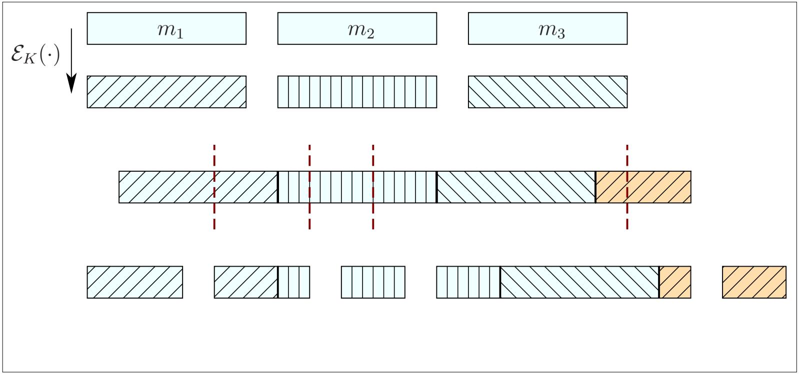



- Then $m_1 \,||\, ¶ \,||\, m_2 \,||\, ¶ \,||\, m_3 \,||\, ¶$ is a prefix of $m'_1 \,||\, m'_2 \,||\, m'_3 \,||\, m'_4 \,||\, m'_5$.

# Correctness Requirement

(explained pictorially)

Royal Holloway University of London Information Security Group

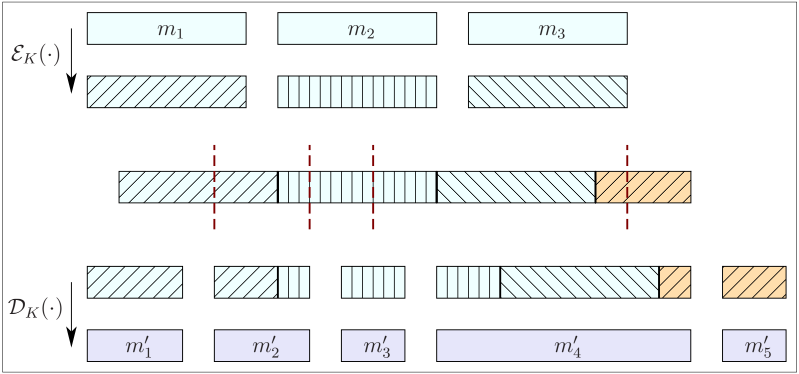



- Then $m_1 \,||\, ¶ \,||\, m_2 \,||\, ¶ \,||\, m_3 \,||\, ¶$ is a prefix of $m'_1 \,||\, m'_2 \,||\, m'_3 \,||\, m'_4 \,||\, m'_5$.

Royal Holloway University of London Information Security Group

# Correctness Requirement

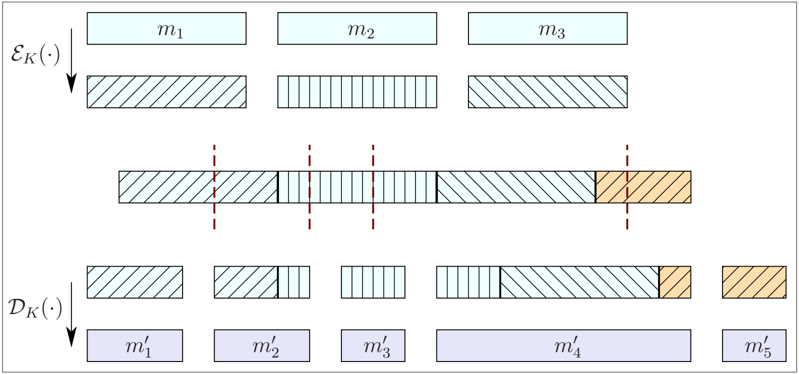
(explained pictorially)




- Then $m_1 \| ¶ \| m_2 \| ¶ \| m_3 \| ¶$ is a prefix of $m'_1 \| m'_2 \| m'_3 \| m'_4 \| m'_5$.

# Correctness Requirement

(explained pictorially)




- Then $m_1 \,||\, \P \,||\, m_2 \,||\, \P \,||\, m_3 \,||\, \P$ is a prefix of $m'_1 \,||\, m'_2 \,||\, m'_3 \,||\, m'_4 \,||\, m'_5$.

# Chosen-Fragment Security

Royal Holloway University of London Information Security Group

- IND-sfCCA **[BKN 04]** extends IND-CCA to protect against **replay** and **out-of-order delivery** attack.
- We extend IND-sfCCA to the fragmented setting, **IND-sfCFA** (Chosen Fragment Attack).
- We provide a **generic construction** for transforming an atomic scheme into a fragmented scheme.
- Starting from an atomic **IND-sfCCA** secure scheme, and a **prefix-free encoding**, the construction gives a fragmented scheme that is **IND-sfCFA** secure.

# Chosen-Fragment Security

- IND-sfCCA **[BKN 04]** extends IND-CCA to protect against **replay** and **out-of-order delivery** attack.
- We extend IND-sfCCA to the fragmented setting, **IND-sfCFA** (Chosen Fragment Attack).
- We provide a **generic construction** for transforming an atomic scheme into a fragmented scheme.
- Starting from an atomic **IND-sfCCA** secure scheme, and a **prefix-free encoding**, the construction gives a fragmented scheme that is **IND-sfCFA** secure.

# End of the Story?

- Our construction shows that Chosen-Fragment Security is not that hard to achieve!
- A closer look at the SSH example, reveals that its designers were aiming for more than just confidentiality.
- We formalize these security goals as: **boundary-hiding** and robustness against **fragmentation-related DoS attacks**.
- Meeting such security goals without compromising confidentiality is more difficult! - as exemplified by the details of the SSH attack.

# Boundary-Hiding

- In the theoretical community it is often regarded as inevitable that a ciphertext leaks the message length. However in practice this is a real problem!

- Practical schemes employ some heuristic techniques in order to protect against **traffic analysis [TV 11]**, **[PRS 11]**, **[DCRS 12]**.

- As we saw earlier SSH encrypts the length field. This does not conceal the message length but can be seen as an attempt to hide ciphertext boundaries.

# Boundary-Hiding

- **BH-CPA** (Informally): Given a concatenation of ciphertexts, no adversary can determine where the ciphertext boundaries lie.

- Correctness requires the decryption algorithm to determine ciphertext boundaries. Thus to achieve boundary-hiding, boundaries should be evident only if the secret key is known.

- We extend our earlier generic construction to also achieve **BH-CPA** by replacing the prefix-free encoding with a **keyed prefix-free encoding**.

- The notion is easily extended to the active setting: **BH-sfCFA**, but is more challenging to achieve.

# Denial of Service

Royal Holloway University of London Information Security Group

- The SSH standard (RFC 4253) suggests limiting the maximum value of the length field in order to mitigate against certain denial-of-service attacks.

- Otherwise an adversary could alter the contents of the length field to indicate a very large value. The receiver would then interpret all subsequent ciphertexts as part of this large ciphertext – **connection hang**.

- Such denial-of-service attacks are not specific to SSH, but to encryption schemes supporting fragmentation in general.

- Informally a scheme is **N-DOS-sfCFA** secure, if no adversary can produce an N-bit long sequence of ciphertext fragments (not output by the encryption oracle) such that the decryption algorithm returns $\varepsilon$ throughout.

# Denial of Service

- The SSH standard (RFC 4253) suggests limiting the maximum value of the length field in order to mitigate against certain denial-of-service attacks.
- Otherwise an adversary could alter the contents of the length field to indicate a very large value. The receiver would then interpret all subsequent ciphertexts as part of this large ciphertext – **connection hang**.
- Such denial-of-service attacks are not specific to SSH, but to encryption schemes supporting fragmentation in general.
- Informally a scheme is **N-DOS-sfCFA** secure, if no adversary can produce an N-bit long sequence of ciphertext fragments (not output by the encryption oracle) such that the decryption algorithm returns $\varepsilon$ throughout.

# Comparing Constructions

| Scheme | IND-sfCFA | BH-CPA | BH-sfCFA | N-DOS-sfCFA $N < \max_{m \in \mathcal{M}}(\lvert m \rvert)$ |
|---|---|---|---|---|
| SSH-CBC | ✘ | ✔ | ✘ | ✘ |
| SSH-CTR | ✔ | ✔ | ✘ | ✘ |
| PF | ✔ | ✘ | ✘ | ✘ |
| KPF | ✔ | ✔ | ✘ | ✘ |
| InterMAC | ✔ | ✔ | ✔ | ✔ |

# Concluding Remarks



- Our work provides a **general framework** for analyzing the security of symmetric encryption schemes over fragmented channels.
- We describe **practical constructions** using **standard primitives**, showing that security in the presence of ciphertext fragmentation can be achieved efficiently and from standard assumptions.
- A full version will be available soon on eprint.