Security of Symmetric Encryption against Mass Surveillance

Overview of an article for the Research Seminar in Cryptography

Kristjan Krips

University of Tartu, Estonia

1 Introduction

This report gives a summary of an article named Security of Symmetric Encryption against Mass Surveillance [BPR14] by Bellare, Paterson and Rogaway. The paper describes if and how it would be possible to subvert the symmetric encryption schemes to enable mass surveillance. Their article shows that this is possible and can be done in the common protocols that are used in the Internet. Besides that, any closed-source software that uses symmetric encryption schemes can be subverted as it is not easy to find out which encryption algorithm is used. The topic is motivated by the recent leaks regarding the surveillance and weakening of cryptographic algorithms by the intelligence agencies [JBG13].

In order to subvert the symmetric encryption schemes the adversary has to replace the original encryption scheme with the subverted encryption scheme but it should be difficult for an observer to distinguish between the two cases. The idea of substituting an algorithm is not new, this method has been described before by Young and Yung [YY97]. The recent leaks show that the intelligence agencies have access to advanced surveillance methods and therefore they may already be using algorithm substation attacks (ASAs).

The ASAs are divided into two, the first attack can be done on the encryption schemes that reveal the initialisation vector (IV) of a block scheme and the second one can be done on any randomised and stateless encryption scheme. The first type of attack describes how the IV can be used to create the subverted encryption scheme and the second type of attack describes how the IV can be chosen such that the ciphertexts can leak the encryption key even when the IV is not known to the adversary. The attack of second type creates a bias in the ciphertext when compared to the non-subverted encryption algorithm and therefore this attack is named the biased-ciphertext attack. However, the authors show that the bias can be designed to be small enough to avoid detection. Finally it is shown how to construct symmetric encryption schemes that resist ASAs.

2 Preliminaries

Notation. A string belongs to $\{0,1\}^*$, $\perp \notin \{0,1\}$ is a special symbol that has the meaning of being invalid. Uniformly sampling x from a set S is denoted by $x \twoheadleftarrow S$.

Syntax. The syntax that is used in the paper supports probabilistic, deterministic or stateful encryption and deterministic or stateful decryption. The encryption algorithms have a parameter named associated data, this is used in order to support authenticated encryption. An encryption scheme is described by a triple $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, where $\mathcal{K}$ marks the key space, $\mathcal{E}$ marks the encryption algorithm and $\mathcal{D}$ marks the decryption algorithm. In the used encryption schemes the key space is finite and not empty. The encryption algorithm $\mathcal{E}$ takes user key, message, associated data and state as input and outputs the ciphertext and the updated state. This is denoted by $(C, \sigma') \twoheadleftarrow \mathcal{E}(K, M, A, \sigma)$. The decryption algorithm $\mathcal{D}$ takes user key, ciphertext, associated data and state as an input and outputs the message and an updated state. This is denoted by $(M, \sigma') \twoheadleftarrow \mathcal{D}(K, C, A, \sigma)$. Both of these algorithms reject if the first member of the output pair is a $\perp$. In the case where an argument to the encryption or decryption algorithm is $\perp$ the output is denoted by a pair $(\perp, \perp)$.

A stateless encryption or decryption algorithm is denoted by using ε as the second component of their output. An encryption scheme Π is stateless if both the encryption and decryption algorithm are stateless. In such case the second component is removed from the output of these algorithms and the state is removed from the parameters of the encryption and decryption algorithm. In the case of the encryption algorithm this means that the algorithm itself generates the required randomness. The paper by Bellare, Paterson and Rogaway shows that in addition to the encryption algorithm being stateful or randomised the decryption algorithm also has to be stateful if we want to avoid ASAs.

Correctness. An encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is correct if $\forall q$ all $M_1, \ldots, M_q \in \{0,1\}^*$ and all $A_1, \ldots, A_q \in \{0,1\}^*$ the following correctness game returns true with probability zero.

$$
\begin{array}{l}
\mathit{Correctness} \\
\left[\begin{array}{l}
\sigma_0 \leftarrow \varepsilon \\
\tau_0 \leftarrow \varepsilon \\
\quad \text{For } i = 1, \ldots, q \text{ do} \\
\quad \left[\begin{array}{l}
(C_i, \sigma_i) \twoheadleftarrow \mathcal{E}(K, M_i, A_i, \sigma_{i-1}) \\
(M_i', \tau_i) \leftarrow \mathcal{D}(K, C_i, A_i, \tau_{i-1})
\end{array}\right. \\
\mathit{Return}\ ((\forall i : C_i \neq \perp) \wedge (\exists i : M_i \neq M_i'))
\end{array}\right.
\end{array}
$$

Security notation The security of symmetric encryption is described with the term privacy. There are two main ways to define the privacy of a symmetric

encryption scheme. We will describe both of them by using the definitions given in [BR05]. We denote a symmetric encryption scheme by a triple $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ and the adversary with $\mathcal{A}$.The goal is to define the privacy by using the indistinguishability under chosen-plaintext attack.

First definition of privacy. The first way to define the privacy is based on the game where the adversary gets access to an encryption oracle such that the adversary can select an input message and either the message is encrypted by the encryption oracle and given to the adversary or the same length message is encrypted and given to the adversary. In our case the adversary can query the oracle a limited number of times and then has to guess if the oracle is encrypting the given input or the a same length message. The encryption scheme is secure under chosen-plaintext attack if such adversary is not successful in finding out which of the messages are encrypted. This is described by defining two types of games, the left game and the right game.

$Left_{\Pi}^{\mathcal{A}}$
$K \twoheadleftarrow \mathcal{K}$
$\sigma \leftarrow \varepsilon$
$b \leftarrow 0$
$b' \leftarrow \mathcal{A}^{Enc}$
$Return(b = b')$

$Enc(M, A)$
$(C, \sigma) \twoheadleftarrow \mathcal{E}(K, 0^{|M|}, A, \sigma)$
$Return\ C$

In the left game the adversary gets access to an oracle who returns the encryption of the message of the same size as the input. In the right game the adversary gets access to an oracle who returns the encryption of the input message.

$Right_{\Pi}^{\mathcal{A}}$
$K \twoheadleftarrow \mathcal{K}$
$\sigma \leftarrow \varepsilon$
$b \leftarrow 1$
$b' \leftarrow \mathcal{A}^{Enc}$
$Return\ (b = b')$

$Enc(M, A)$
$(C, \sigma) \twoheadleftarrow \mathcal{E}(K, M, A, \sigma)$
$Return\ C$

The ind-cpa advantage of such adversary is defined by the adversary's ability to find out which game he is playing, i.e., which oracle he is given. This is defined as

$$Adv_{\Pi}^{ind-cpa}(\mathcal{A}) = Pr[Right_{\Pi}^{\mathcal{A}} = true] - Pr[Left_{\Pi}^{\mathcal{A}} = true]$$

Second definition of privacy. For the second definition we give the adversary a modified task which is defined by the game PRIV.

$PRIV_{\Pi}^{\mathcal{A}}$

$K \leftarrow\!\!\!\!\!\!\$\ \mathcal{K}$
$\sigma \leftarrow \varepsilon$
$b \leftarrow\!\!\!\!\!\!\$\ \{0,1\}$
$b' \leftarrow \mathcal{A}^{Enc}$
Return $(b = b')$

$Enc(M, A)$

If $b = 1$
  $(C, \sigma) \leftarrow\!\!\!\!\!\!\$\ \mathcal{E}(K, M, A, \sigma)$
Else
  $(C, \sigma) \leftarrow\!\!\!\!\!\!\$\ \mathcal{E}(K, 0^{|M|}, A, \sigma)$
Return C

In this game the adversary has to guess the world he is in. In this case the advantage of the adversary measures how much better than a random guess can the adversary do. We will name this the privacy advantage of the adversary and define it as

$$Adv_{\Pi}^{priv}(\mathcal{A}) = 2 \cdot Pr[PRIV_{\Pi}^{\mathcal{A}} = true] - 1$$

When we look at the game PRIV then we see that in the case where $b = 0$ we have the game $Left_{\Pi}^{\mathcal{A}}$ and in the case where $b = 1$ we have the game $Right_{\Pi}^{\mathcal{A}}$. Therefore, $Pr[PRIV_{\Pi}^{\mathcal{A}} = true]$ gives the probability of the adversary $\mathcal{A}$ correctly guessing which game is he playing. Now we show how the advantage of the adversary is found.

Proof.

$$\begin{aligned}
Pr[PRIV_{\Pi}^{\mathcal{A}} = true] &= Pr[b = b'] \\
&= Pr[b = b'|b = 1] \cdot Pr[b = 1] + Pr[b = b'|b = 0] \cdot Pr[b = 0] \\
&= Pr[b = b'|b = 1] \cdot \frac{1}{2} + Pr[b = b'|b = 0] \cdot \frac{1}{2} \\
&= Pr[b' = 1|b = 1] \cdot \frac{1}{2} + Pr[b' = 0|b = 0] \cdot \frac{1}{2} \\
&= Pr[b' = 1|b = 1] \cdot \frac{1}{2} + (1 - Pr[b' = 1|b = 0]) \cdot \frac{1}{2} \\
&= \frac{1}{2} + \frac{1}{2} \cdot (Pr[b' = 1|b = 1] - Pr[b' = 1|b = 0]) \\
&= \frac{1}{2} + \frac{1}{2} \cdot (Pr[Right_{\Pi}^{\mathcal{A}} = true] - Pr[Left_{\Pi}^{\mathcal{A}} = true]) \\
&= \frac{1}{2} + \frac{1}{2} \cdot Adv_{\Pi}^{ind-cpa}(\mathcal{A})
\end{aligned}$$

Therefore, $\frac{1}{2} \cdot Adv_{\Pi}^{priv}(\mathcal{A}) = Pr[PRIV_{\Pi}^{\mathcal{A}} = true] - \frac{1}{2}$, which is equal to the proposed advantage if we multiply both sides by two.

3 Subverting Symmetric Encryption

We denote the subverted encryption algorithm with the letter $\tilde{\mathcal{E}}$. The subverted encryption algorithm takes as an additional input the key $\tilde{K}$ of the big-brother.

The additional key $\tilde{K}$ can be embedded in the subverted code. It is important to note that the adversary, i.e., the big-brother has to hide the attack and therefore subversion solutions which the user may detect can not be used. Therefore, the adversary has to design an encryption scheme where the ciphertexts look like the real ones but where the message or the user key can be revealed. In the following it is required that the ciphertexts created by the subverted encryption schemes decrypt normally under the real decryption algorithm. This requirement is needed to avoid the detection of the attack. In order to formalise the detectability and success of the ASA the terms detection security and surveillance security are introduced. Detection security requires that the real ciphertexts and subverted ciphertexts should be indistinguishable for a testing adversary who has access to the user keys but does not have access to the key $\tilde{K}$ of the big-brother. Surveillance security requires that the big-brother can not distinguish real ciphertexts and subverted ciphertexts even with access to the key $\tilde{K}$. Both of these formalisations support multiple users but only one big-brother.

Subversions. A subversion of an encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is denoted by $\tilde{\Pi} = (\tilde{\mathcal{K}}, \tilde{\mathcal{E}}, \tilde{\mathcal{D}})$, where the key space $\tilde{\mathcal{K}}$ is finite and nonempty. More specifically, the subverted encryption algorithm $\tilde{E}$ can be randomised, stateful and is denoted by $(C, \sigma') \leftarrow\!\!\$\ \tilde{\mathcal{E}}(\tilde{\mathcal{K}}, K, M, A, \sigma, i)$. In addition to the big-brother key, the subverted encryption algorithm takes as an input information that uniquely identifies the user (e.g., IP aadress, MAC address) who is using the key K, this information is denoted by the letter i. The big-brother has an algorithm denoted by $\tilde{D}$ that is able to recover the plaintext. This algorithm is denoted by $\tilde{\mathcal{D}}(\tilde{K}, \mathbf{C}, \mathbf{A}, i)$, where the parameters are the big-brother key, a vector of ciphertexts, a vector of associated data and the identifier of the user, i.e., the owner of the key K. The plaintext recovery algorithm may use different methods for finding the plaintext, e.g., it might first do a key recovery and then find the plaintexts. It is important to note that the subverted encryption scheme satisfies the decryptability condition relative to the initial encryption scheme if it is a correct encryption scheme $(\tilde{\mathcal{K}} \times \mathcal{K}, \tilde{E}, \mathcal{D}')$, where the modified decryption algorithm is defined as $\mathcal{D}'((\tilde{\mathcal{K}}, \mathcal{K}), C, A, \sigma) = \mathcal{D}(\mathcal{K}, C, A, \sigma)$. Therefore, the user can decrypt the subverted ciphertexts using the non-subverted decryption algorithm. In order to describe the security requirements of the subverted encryption scheme the detection advantage and surveillance advantage is found.

Detection advantage. The detection advantage shows how well the users are able to distinguish the non-subverted encryption schemes and the subverted encryption schemes. In order to find the detection advantage a security game named DETECT is defined. In this game a detection test named $\mathcal{U}$ is used to distinguish the subverted encryption scheme from a non-subverted encryption scheme.

$DETECT^{\mathcal{U}}_{\Pi,\tilde{\Pi}}$
$b \twoheadleftarrow \{0,1\}$
$\tilde{K} \twoheadleftarrow \tilde{\mathcal{K}}$
$b' \leftarrow \mathcal{U}^{Key,Enc}$
$Return\ (b = b')$

$Key(i)$
If $K_i = \perp$
  $K_i \twoheadleftarrow \mathcal{K}$
  $\sigma_i \leftarrow \varepsilon$
$Return\ K_i$

$Enc(M, A, i)$
If $K_i = \perp$
  $Return\ \perp$
If $b = 1$
  $(C, \sigma_i) \twoheadleftarrow \mathcal{E}(K_i, M, A, \sigma_i)$
Else
  $(C, \sigma_i) \twoheadleftarrow \tilde{\mathcal{E}}(\tilde{K}, K_i, M, A, \sigma_i, i)$
$Return\ C$

Now we can write the advantage of the detection test $\mathcal{U}$ as

$$Adv^{det}_{\Pi,\tilde{\Pi}}(\mathcal{U}) = 2 \cdot Pr[DETECT^{\mathcal{U}}_{\Pi,\tilde{\Pi}} = true] - 1.$$

If the advantage of the detection tests is negligible for the given subversion then the subversion is undetectable. In such case the big-brother would be successful as this would mean that the he will not be detected by testing the closed source implementation of an encryption scheme. Thus, one would have to reverse engineer the implementation to find out if the encryption scheme is subverted.

Surveillance advantage. The surveillance advantage shows how secure an encryption scheme can be against the ASAs. The idea is to let the big-brother to distinguish between ciphertexts produced by the subverted encryption algorithm and the non-subverted encryption algorithm. It is important to notice that while doing so the big-brother has access to the master-key $\tilde{K}$. In the following we define a surveillance game named SURV, where $\mathcal{B}$ denotes the big-brother who is trying to distinguish the two cases.

$SURV^{\mathcal{B}}_{\Pi,\tilde{\Pi}}$
$b \twoheadleftarrow \{0,1\}$
$\tilde{K} \twoheadleftarrow \tilde{\mathcal{K}}$
$b' \leftarrow \mathcal{B}^{Key,Enc}(\tilde{K})$
$Return\ (b = b')$

$Key(i)$
If $K_i = \perp$
  $K_i \twoheadleftarrow \mathcal{K}$
  $\sigma_i \leftarrow \varepsilon$
$Return\ \varepsilon$

$Enc(M, A, i)$
If $K_i = \perp$
  $Return\ \perp$
If $b = 1$
  $(C, \sigma_i) \twoheadleftarrow \mathcal{E}(K_i, M, A, \sigma_i)$
Else
  $(C, \sigma_i) \twoheadleftarrow \tilde{\mathcal{E}}(\tilde{K}, K_i, M, A, \sigma_i, i)$
$Return\ C$

Now we can write the advantage of the big-brother $\mathcal{B}$ in the surveillance game SURV as

$$Adv^{srv}_{\Pi,\tilde{\Pi}}(\mathcal{B}) = 2 \cdot Pr[SURV^{\mathcal{B}}_{\Pi,\tilde{\Pi}} = true] - 1.$$

If the advantage of any big-brother in the surveillance game is negligible for the given encryption scheme Π and for any subversion $\tilde{\Pi}$ then the encryption scheme Π is secure against surveillance. In the following only one user is considered in the surveillance games as this will simplify the proofs. The advantage against the one user game compared to multi-user games can grow by a factor of the number of users.

4 Algorithm Substitution Attacks

This section describes two ways to implement ASAs. The first attack shows how the encryption modes that surface the IV can be subverted by hiding the user key into the IV. The second kind of an attack shows that if the IV does not surface then it is possible to leak parts of the user by constructing a biased ciphertext. These results show that randomised and stateless encryption schemes are not secure against ASAs.

4.1 ASA by IV replacement

Let there be a randomised, stateless encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, where $C \leftarrow \mathcal{E}(K, M, A, IV)$. The encryption scheme Π surfaces the IV if there is an efficient algorithm χ that outputs the IV from a ciphertext, $\chi(\mathcal{E}(K, M, A, IV)) = IV$. The encryption modes that surface the IV are e.g., CBC and CTR.

Stateful ASA. In this attack the IV is replaced by the encryption of the user key. If the user key length extends the length of the IV then the user key could be leaked over several ciphertexts. If the user key is shorter than then IV then padding could be used. However, to simplify the attack we assume that the key length is equal to the length of the IV. In the detection test it should not be possible to distinguish between the real encryption scheme and the subverted encryption scheme and therefore the IV has to be unique. Thus, the encryption scheme has to be stateful if the big-brother wants to avoid detection.

In the following we describe the subversion of an encryption scheme Π, which is denoted by $\tilde{\Pi} = (\tilde{\mathcal{K}}, \tilde{\mathcal{E}}, \tilde{\mathcal{D}})$. Let the bit length of the IV and user key be n. In order to let the big-brother encrypt the user key we have to have a blockcipher $E : \tilde{\mathcal{K}} \times \{0,1\}^n \rightarrow \{0,1\}^n$. Now we can specify the subverted encryption algorithm and the plaintext recovery algorithm. It is important to notice that the subverted encryption algorithm reveals the user key only for the initial state. The letters that are written in bold denote a vector, i.e., in the description of the plaintext recovery algorithm $\mathbf{C}$ denotes a vector of ciphertexts and $\mathbf{A}$ denotes a vector of associated data.

$\tilde{\mathcal{E}}(\tilde{K}, K, M, A, \sigma, i)$

$\left\lceil \begin{array}{l} \text{If } \sigma = 0 \\ \quad \left[IV \leftarrow E(\tilde{K}, K) \right. \\ \text{Else} \\ \quad \left[IV \twoheadleftarrow \{0,1\}^n \right. \\ C \leftarrow \mathcal{E}(K, M, A, IV) \\ \sigma \leftarrow \sigma + 1 \\ \textit{Return } C \end{array} \right.$

$\tilde{\mathcal{D}}(\tilde{K}, \mathbf{C}, \mathbf{A}, \sigma, i)$

$\left\lceil \begin{array}{l} IV \leftarrow \chi(\mathbf{C}[1]) \\ K \leftarrow E^{-1}(\tilde{K}, IV) \\ \mathbf{M}[1] \leftarrow \mathcal{D}(K, \mathbf{C}[1], \mathbf{A}[1]) \\ \textit{Return } \mathbf{M} \end{array} \right.$

The subverted IV is indistinguishable from a random IV if the blockcipher E is a PRP/PRF. This is formally stated in the paper [BPR14] with the following theorem.

Theorem 1. *Let $\Pi = (\{0,1\}^n, \mathcal{E}, \mathcal{D})$ be a randomised, stateless symmetric encryption scheme that surfaces an IV of length n. Let $E : \tilde{\mathcal{K}} \times \{0,1\}^n \rightarrow \{0,1\}^n$ be a blockcipher. Let the subversion $\tilde{\Pi} = (\tilde{\mathcal{K}}, \tilde{\mathcal{E}}, \tilde{\mathcal{D}})$ of Π be defined as above. Let $\mathcal{U}$ be a test that makes q queries to its Key oracle. Then we can construct an adversary $\mathcal{A}$ such that $Adv^{det}_{\Pi,\tilde{\Pi}}(\mathcal{U}) \leq \frac{q^2}{2^n} + Adv^{prf}_{E}(\mathcal{A})$. Adversary $\mathcal{A}$ makes q oracle queries and its running time is that of $\mathcal{U}$.*

The term $\frac{q^2}{2^n}$ comes because there might be two users with the same key and in this case the IV will be the same because the subverted IV is an encryption of the user key. Still, this simple attack has a drawback, namely when the user system is reset and the encryption software is reinstalled then the state is also reset and thus the subverted IV is recreated. This problem is solved in the next paragraph by using a stateless and randomised encryption scheme.

Stateless ASA. Let k be the key length of the encryption scheme Π and let $v = \lceil log_2(k) \rceil$. The blockcipher is described by $E : \tilde{\mathcal{K}} \times \{0,1\}^n \rightarrow \{0,1\}^n$, where n is the length of the IV. The subversion of the encryption scheme $\mathcal{E}$ is a triple $\tilde{\Pi} = (\tilde{\mathcal{K}}, \tilde{\mathcal{E}}, \tilde{\mathcal{D}})$, where

$\tilde{\mathcal{E}}(\tilde{K}, K, M, A, i)$

$\left\lceil \begin{array}{l} l \twoheadleftarrow [1..k] \\ R \twoheadleftarrow \{0,1\}^{n-v-1} \\ IV \twoheadleftarrow\!\leftarrow E(\tilde{K}, K[l]||l||R) \\ C \leftarrow \mathcal{E}(K, M, A, IV) \\ \textit{Return } C \end{array} \right.$

$\tilde{\mathcal{D}}(\tilde{K}, \mathbf{C}, \mathbf{A}, i)$

$\left\lceil \begin{array}{l} \text{For } j = 1, \ldots, |\mathbf{C}| \text{ do} \\ \quad \left\lceil b||l||R \leftarrow E^{-1}(\tilde{K}, \chi(\mathbf{C}[j])) \right. \\ \quad \left\lfloor K'[l] \leftarrow b \right. \\ \text{For } j = 1, \ldots, |\mathbf{C}| \text{ do} \\ \quad \left[\mathbf{M}[j] \leftarrow \mathcal{D}(K', \mathbf{C}[j], \mathbf{A}[j]) \right. \\ \textit{Return } \mathbf{M} \end{array} \right.$

In these algorithms the letter l denotes an integer that is encoded as a bit-string of length v. The undetectability of the subversion depends on the used blockcipher, i.e., it depends on the security of the PRP/PRF as the key $\tilde{K}$ is only known to the big-brother. The big-brother will succeed if he will be able to get access to at least $k \cdot ln(k)$ ciphertexts.

Theorem 2. *Let $\Pi = (\{0,1\}^k, \mathcal{E}, \mathcal{D})$ be a randomised, stateless symmetric encryption scheme that surfaces an IV of length n. Let $E : \tilde{\mathcal{K}} \times \{0,1\}^n \to \{0,1\}^n$ be a blockcipher. Let $v = \lceil log_2(k) \rceil$. Let the subversion $\tilde{\Pi} = (\tilde{\mathcal{K}}, \tilde{\mathcal{E}}, \tilde{\mathcal{D}})$ of Π be defined as above. Let $\mathcal{U}$ be a test that makes q queries to its Enc oracle. Then we can construct an adversary $\mathcal{A}$ such that $Adv^{det}_{\Pi,\tilde{\Pi}}(\mathcal{U}) \leq q^2/2^{n-v-1} + Adv^{prf}_{E}(\mathcal{A})$. Adversary $\mathcal{A}$ makes q oracle queries and its running time is that of $\mathcal{U}$.*

In this case there is no state and therefore the detection advantage does not depend on the resets of the system.

4.2 The biased ciphertext attack

The previous attacks can not be applied on some protocols and therefore we need another approach for them. The following will describe how it is possible to subvert any randomised and stateless encryption scheme that uses a small amount of randomness. The biased ciphertext attack is based on the idea to create an encryption scheme that creates biased randomness and this randomness is used to create the ciphertext. Therefore, the ciphertext of the subverted encryption scheme is biased compared to the non-subverted encryption scheme and as the big-brother knows how the bias is created then he can also find the information that is hidden in the ciphertexts. In this attack the big-brother has to capture several ciphertexts in order to get access to the user key as from each ciphertext only one bit of the user key can be recovered.

Let D be the space of randomness and let δ denote the random coins that will be used in the encryption scheme. In the following δ will be used instead of IV as IV might not surface. In order to encode the key bit a specific randomness has to be used. Let $g(\cdot) \leftarrow \mathcal{E}(K, M, A, \cdot)||\sigma||i$ be a function that only takes randomness as input. Also, let $g : D \to R$, where $D \subseteq \{0,1\}^*$ and $f : \{0,1\}^* \to \{0,1\}$. Now, to choose the biased randomness we have to create a set with the valid random coins, this is done by $S^{f,g}(b, D) = \{\delta \in D : f(g(\delta)) = b\}$, where $b \in \{0,1\}$. Let F be a PRF, then the key bit can be found from the ciphertext by $F(\tilde{K}, C) = K[j]$ if $K[j]$ was encoded into C. For the attack to succeed it is required that the encryption scheme is coin injective, i.e., the mapping of random coins to ciphertexts is injective for each fixed key, message and associated data.

The subversion of the encryption scheme is described with the following subverted encryption and subverted decryption algorithm. In order to recover the user key at least $|K|$ ciphertexts have to be captured by the adversary.

$\tilde{\mathcal{E}}(\tilde{K}, K, M, A, \sigma, i)$

$j \leftarrow \sigma \bmod |K|$
$j \leftarrow j + 1$
$g(\cdot) \leftarrow \mathcal{E}(K, M, A, \cdot)||\sigma||i$
$\delta \twoheadleftarrow S^{F(\tilde{K},\cdot),g(\cdot)}(K[j], D)$
$C \leftarrow \mathcal{E}(K, M, A, \delta)$
$\sigma \leftarrow \sigma + 1$
Return C

$\tilde{\mathcal{D}}(\tilde{K}, \mathbf{C}, \mathbf{A}, i)$

For $j = 1, \ldots, |\mathbf{C}|$ do
  $K'[j] \leftarrow F(\tilde{K}, \mathbf{C}[j]||j-1||i)$
For $j = 1, \ldots, |\mathbf{C}|$ do
  $\mathbf{M}[j] \leftarrow \mathcal{D}(K', \mathbf{C}[j], \mathbf{A}[j])$
Return $\mathbf{M}$

The described subversion is undetectable. In order to show that a lemma is given. The proof of the lemma can be found in the paper [BPR14].

Lemma 1. *Suppose* $g : D \to R$. *Let* $b \in \{0,1\}$ *and* $\overline{\delta} \in D$. *Let* $d = |D|$. *Let* $p = Pr[\delta = \overline{\delta}]$ *where we first draw* $f : g(D) \to \{0,1\}$ *at random and then draw* δ *at random from* $S^{f,g}(b, D) = \{\delta \in D : f(g(\delta)) = b\}$
(1) If g *is injective then* $p = (1-2)^{-d})/d$.
(2) More generally, if g *is* k*-regular, then* $p = (1 - 2^{-d/k})/d$.

Theorem 3. *Let* $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ *be a randomised, stateless, coin-injective symmetric encryption scheme with randomless-length* r, *and let* $d = 2^r$. *Let* $F : \tilde{\mathcal{K}} \times \{0,1\}^n \to \{0,1\}^n$ *be a PRF. Let the subversion* $\tilde{\Pi} = (\tilde{\mathcal{K}}, \tilde{\mathcal{E}}, \tilde{\mathcal{D}})$ *of* Π *be defined as above. Let* $\mathcal{U}$ *be a test that makes* q *queries to its Enc oracle. Then we can construct an adversary* $\mathcal{A}$ *such that* $Adv^{det}_{\Pi,\tilde{\Pi}}(\mathcal{U}) \leq q^2/2^d + Adv^{prf}_{E}(\mathcal{A})$. *Adversary* $\mathcal{A}$ *makes* q *oracle queries and its running time is that of* $\mathcal{U}$.

Therefore, if sufficient amount of randomness is used then the subversion is undetectable. The hints for the proof can be found in the paper [BPR14]. If the system is restarted and thus the state is reset then the detection becomes more probable.

5 Defeating Algorithm Substitution Attacks

It is possible to create encryption schemes that efficiently resist the ASAs. This section will describe how such encryption schemes can be constructed. We know from the previous sections that an encryption scheme has to be deterministic and stateful to prevent the ASAs. We will see that one requirement for such encryption schemes is that they have to have unique ciphertexts. I.e., for any key, message, associated data and state there must be only one ciphertext that can be decrypted into the corresponding message. Any unique ciphertext scheme is deterministic due to the correctness requirement.

Theorem 4. *Let* $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ *be a unique ciphertext symmetric encryption scheme. Let* $\tilde{\Pi} = (\tilde{\mathcal{K}}, \tilde{\mathcal{E}}, \tilde{\mathcal{D}})$ *be a subversion of* Π *that obeys the decryptability condition relative to* Π. *Let* $\mathcal{B}$ *be an adversary. Then* $Adv^{src}_{\Pi,\tilde{\Pi}}(\mathcal{B}) = 0$.

The proof of the theorem is done by using induction and it can be found in the paper [BPR14].

An example of a unique-ciphertext scheme. The following describes a scheme with unique ciphertexts. It is based on encode-then-encipher paradigm from [BR00]. Let $P: \{0,1\}^k \times \{0,1\}^* \to \{0,1\}^*$ be a family of permutations. Therefore, P_K is injective and length-preserving for every key. The inverse of P is denoted by P^{-1}. In addition, let $F: \{0,1\}^k \times \{0,1\}^* \to \{0,1\}^t$ be a family of functions that will be used for generating the MAC. The state is denoted by σ and in the following scheme it is a counter which can be represented in (l-bit) string form by $\langle\sigma\rangle$. The key space of the following scheme is $\mathcal{K} = \{0,1\}^{2k}$. Now we can define the encryption and decryption algorithms for the unique ciphertext scheme.

$\mathcal{E}(K, M, A, \sigma)$
- If $\sigma = 2^l$
  - *Return* $(\perp, \sigma)$
- $K_1||K_2 \leftarrow K$
- $W \leftarrow P(K_1, \langle\sigma\rangle||M$
- $T \leftarrow F(K_2, W||A)$
- $C \leftarrow (W, T)$
- $\sigma \leftarrow \sigma + 1$
- *Return* (C, σ)

$\mathcal{D}(K, C, A, \tau)$
- If $\tau = 2^l$
  - *Return* $(\perp, \tau)$
- $K_1||K_2 \leftarrow K$
- $(W, T) \leftarrow C$
- $x \leftarrow P^{-1}(K_1, W)$
- If $|x| < l$
  - *Return* $(\perp, \tau)$
- $\langle\sigma\rangle||M \leftarrow x$
- If $T \neq F(K_2, W||A))$
  - *Return* $(\perp, \tau)$
- If $\sigma \neq \tau)$
  - *Return* $(\perp, \tau)$
- $\tau \leftarrow \tau + 1$
- *Return* (M, τ)

Theorem 5. *Let $P: \{0,1\}^k \times \{0,1\}^* \to \{0,1\}^*$ be a family of permutations and $F: \{0,1\}^k \times \{0,1\}^* \to \{0,1\}^t$ a family of functions. Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be the symmetric encryption scheme associated to them as above. Then Π satisfies the correctness condition and has unique ciphertexts.*

The proof of the theorem can be found in the paper [BPR14].

Surveillance-resistance from nonce-based schemes. Any nonce-based scheme that has the non-degeneracy condition can be turned into a stateful symmetric encryption scheme that has unique ciphertexts. This can be done by using the nonce as a counter. A nonce-based encryption scheme is a triple $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, where $\mathcal{K}$ is finite and nonempty, $\mathcal{E}$ is a deterministic algorithm such that $\mathcal{E}(K, M, A, N) = C$ and $\mathcal{D}$ is a deterministic algorithm such that $\mathcal{D}(K, C, A, N) \in \{0,1\}^* \cup \{\perp\}$. The scheme has to be correct. A stateful encryption scheme $\Pi^* = (\mathcal{K}^*, \mathcal{E}^*, \mathcal{D}^*)$

can be associated with the nonce based scheme in the following way. The keyspace of the corresponding stateful encryption scheme is the same as in the nonce based scheme. Now the encryption and decryption algorithms can be defined.

$\mathcal{E}^*(K, M, A, \sigma)$
$\big[$ *Return* $(\mathcal{E}(K, M, A, \sigma), \sigma + 1)$

$\mathcal{D}^*(K, C, A, \tau)$
$\big[$ *Return* $(\mathcal{D}(K, C, A, \tau), \tau + 1)$

The nonce-based scheme is non-degenerate if there is at most one ciphertext for every possible K,M,A,N such that $\mathcal{D}(K, C, A, N) = M$.

Theorem 6. *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a non-degenerate nonce-based scheme and let $\Pi^* = (\mathcal{K}^*, \mathcal{E}^*, \mathcal{D}^*)$ be the stateful symmetric encryption scheme obtained from Π as above. Then Π^* satisfies the correctness condition and has unique ciphertexts.*

6 Is Subversion Possible in the Standard Internet Protocols

Bellare, Paterson and Rogaway found that several commonly used Internet protocols are vulnerable to ASAs. In this section we give a summary of their findings in a compact form. For the details about the subversion of standard internet protocols, see [BPR14].

Protocol	**Mode**	**Surfaces IV**	**IV replacement attack**	**Biased ciphertext attack**
SSL 3.0	CBC	no	not vulnerable	vulnerable
TLS 1.0	CBC	no	not vulnerable	vulnerable
TLS 1.1	CBC	yes	vulnerable	-
TLS 1.2	CBC	yes	vulnerable	-
IPsec	CBC	yes	vulnerable	-
IPsec	AES-GCM	yes	vulnerable	-
IPsec	AES-CCM	yes	vulnerable	-
SSH	CBC	no	not vulnerable	vulnerable
SSH	CTR	no	not vulnerable	vulnerable
SSH	AES-GCM	no	not vulnerable	vulnerable

7 Asymmetric Subversion

The big-brother might want to avoid the situation where the master key is found by third parties as this would give them the ability do to the surveillance themselves. This is a real possibility if the big-brother uses a symmetric encryption key and the key is embedded in the code of the subverted encryption algorithm.

In such case a third party could reverse engineer the implementation and find the master key. In order to avoid this the big-brother might use asymmetric encryption an embed a public key into the encryption code. This is described by Young and Yung in [YY97].

In order to use asymmetric encryption the master key $\tilde{K}$ is replaced by $(\tilde{K}, \tilde{L})$, where $\tilde{K}$ is the public key and $\tilde{L}$ is private key. In this case the subverted encryption algorithm does not change, it remains $(C, \sigma_i) \twoheadleftarrow \tilde{\mathcal{E}}(\tilde{K}, K, M, A, \sigma, i)$ but the plaintext recovery algorithm has to be changed to use the private key instead. Now the modified detection game and the modified surveillance games can be defined.

Detection2. The following detection game describes the situation where the big-brother uses asymmetric encryption.

$DETECT2^{\mathfrak{U}}_{\Pi,\tilde{\Pi}}$
- $b \twoheadleftarrow \{0,1\}$
- $(\tilde{K}, \tilde{L}) \twoheadleftarrow \tilde{\mathcal{K}}$
- $b' \leftarrow \mathfrak{U}^{Key,Enc}(\tilde{K})$
- *Return* $(b = b')$

$Key(i)$
- If $K_i = \perp$
  - $K_i \twoheadleftarrow \mathcal{K}$
  - $\sigma_i \leftarrow \varepsilon$
- *Return* K_i

$Enc(M, A, i)$
- If $K_i = \perp$
  - *Return* $\perp$
- If $b = 1$
  - $(C, \sigma_i) \twoheadleftarrow \mathcal{E}(K_i, M, A, \sigma_i)$
- Else
  - $(C, \sigma_i) \twoheadleftarrow \tilde{\mathcal{E}}(\tilde{K}, K_i, M, A, \sigma_i, i)$
- *Return* C

Now we can write the advantage of the detection test $\mathfrak{U}$ as

$$Adv^{det2}_{\Pi,\tilde{\Pi}}(\mathfrak{U}) = 2 \cdot Pr[DETECT2^{\mathfrak{U}}_{\Pi,\tilde{\Pi}} = true] - 1.$$

Surveillance2. The following surveillance game describes the situation where the big-brother uses asymmetric encryption.

$SURV2^{\mathfrak{B}}_{\Pi,\tilde{\Pi}}$
- $b \twoheadleftarrow \{0,1\}$
- $(\tilde{K}, \tilde{L}) \twoheadleftarrow \tilde{\mathcal{K}}$
- $b' \leftarrow \mathfrak{B}^{Key,Enc}(\tilde{L})$
- *Return* $(b = b')$

$Key(i)$
- If $K_i = \perp$
  - $K_i \twoheadleftarrow \mathcal{K}$
  - $\sigma_i \leftarrow \varepsilon$
- *Return* ε

$Enc(M, A, i)$
- If $K_i = \perp$
  - *Return* $\perp$
- If $b = 1$
  - $(C, \sigma_i) \twoheadleftarrow \mathcal{E}(K_i, M, A, \sigma_i)$
- Else
  - $(C, \sigma_i) \twoheadleftarrow \tilde{\mathcal{E}}(\tilde{K}, K_i, M, A, \sigma_i, i)$
- *Return* C

Now we can write the advantage of the big-brother $\mathcal{B}$ in the surveillance game SURV as $Adv^{srv2}_{\Pi,\tilde{\Pi}}(\mathcal{B}) = 2 \cdot Pr[SURV2^{\mathcal{B}}_{\Pi,\tilde{\Pi}} = true] - 1$.

References

[BPR14] Mihir Bellare, KennethG. Paterson, and Phillip Rogaway. Security of symmetric encryption against mass surveillance. In JuanA. Garay and Rosario Gennaro, editors, *Advances in Cryptology – CRYPTO 2014*, volume 8616 of *Lecture Notes in Computer Science*, pages 1–19. Springer Berlin Heidelberg, 2014.

[BR00] Mihir Bellare and Phillip Rogaway. Encode-then-encipher encryption: How to exploit nonces or redundancy in plaintexts for efficient cryptography. In Tatsuaki Okamoto, editor, *Advances in Cryptology ASIACRYPT 2000*, volume 1976 of *Lecture Notes in Computer Science*, pages 317–330. Springer Berlin Heidelberg, 2000.

[BR05] Mihir Bellare and Phillip Rogaway. Introduction to modern cryptography, chapter 4: symmetric encryption, 2005. `https://cseweb.ucsd.edu/~mihir/cse207/w-se.pdf`.

[JBG13] Julian Borger James Ball and Glenn Greenwald. Revealed: how us and uk spy agencies defeat internet privacy and security. The Guardian, September 2013. `http://www.theguardian.com/world/2013/sep/05/nsa-gchq-encryption-codes-security`.

[YY97] Adam Young and Moti Yung. Kleptography: Using cryptography against cryptography. In Walter Fumy, editor, *Advances in Cryptology EUROCRYPT 97*, volume 1233 of *Lecture Notes in Computer Science*, pages 62–74. Springer Berlin Heidelberg, 1997.