Preliminary versions of this paper appear in *2007 IEEE Symposium on Security & Privacy Proceedings*, pp. 92–100, 2007 and in *IET Information Security Journal*, vol. 5, no. 4, pp. 207–219, 2011. This is the full version.

Provable-Security Analysis of Authenticated Encryption in Kerberos

Alexandra Boldyreva Virendra Kumar
Georgia Institute of Technology, School of Computer Science
266 Ferst Drive, Atlanta, GA 30332-0765 USA
{sasha,virendra}@gatech.edu

Abstract

Kerberos is a widely deployed network authentication protocol currently being considered for standardization. Many works have analyzed its security, identifying flaws and often suggesting fixes, thus promoting the protocol's evolution. Several recent results present successful, formal methods-based verifications of a significant portion of the current version, v.5, and some even imply security in the computational setting. For these results to hold, encryption in Kerberos should satisfy strong cryptographic security notions. However, prior to our work, none of the encryption schemes currently deployed as part of Kerberos, nor their proposed revisions, were known to provably satisfy such notions. We take a close look at Kerberos' encryption, and we confirm that most of the options in the current version provably provide privacy and authenticity, though some require slight modifications which we suggest. Our results complement the formal methods-based analysis of Kerberos that justifies its current design.

Keywords: Kerberos, authenticated encryption, provable security.

1 Introduction

1.1 Motivation

Kerberos is a trusted third party network authentication protocol. It allows a client to authenticate herself to multiple services, e.g. file servers and printers, with a single login. Kerberos has become widely deployed since its origination as MIT's project Athena in 1988. It has been adopted by many large universities and corporations, is part of all major computing platforms, e.g. Windows (starting from Windows 2000), Linux, UNIX, and Mac OS X, and is a draft standard at IETF [30].

Security of Kerberos has been analyzed in many works, e.g. [16, 29, 6, 5, 27, 20, 31]. Most commonly, analyses identify certain limitations or flaws in the deployed versions of Kerberos and sometimes propose fixes. This leads to the evolution of the protocol, when a new version patches

the known vulnerabilities of the previous versions. The current version, Kerberos v.5, is already being revised and extended [24, 26, 25].

What is certainly desirable for the upcoming standard is to provide some guarantees that the protocol not only resists some *specific* known attacks, but that it withstands a very large class of possible attacks, under some well-studied assumptions. Modern techniques in cryptography (computational approach), and formal methods (symbolic approach) make it possible; however, formally analyzing such a complex protocol is not an easy task.

Several recent works contributed in this direction. Butler et al. [17, 18] have analyzed significant portions of the current version of Kerberos and its extensions in the symbolic approach (i.e. Dolev-Yao model [19]), and have formally verified that the design of Kerberos' current version meets the desired goals for the most part. However, a known limitation of such analyses is a high level of abstraction. A significant advance has been made by a recent work by Backes et al. [1] in that it is the only work providing symbolic analysis that also guarantees security in the computational setting, the well-accepted strongest model of security. Their results use the computational soundness model due to Backes et al. [4, 3, 2]. However, for their results to hold, cryptographic primitives used in the protocol need to satisfy strong notions of security (in the computational setting). Namely, as the result of [23] implies, the encryption scheme utilized by the protocol needs to provide privacy against chosen-ciphertext attacks (be IND-CCA secure), as well as authenticity and integrity of ciphertexts (be INT-CTXT secure).

However, it is not known whether *authenticated* encryption[1] in Kerberos is IND-CCA and INT-CTXT secure. Certain known vulnerabilities indicate that encryption in version 4 did not satisfy these notions [31]. While encryption in the current version, v.5, is designed to resist known attacks, it is not clear whether it *provably* resists all attacks of the class, and if it does, under what assumptions. Provable security has become a de facto standard approach in modern cryptography research. Cryptographers design plenty of cryptographic schemes for a vast range of possible applications, and they usually provide rigorous proofs of security for their constructions. It is somewhat surprising then that the schemes that are actually used in deployed protocols remain unanalyzed from the provable security perspective. Our work aims to close this gap.

1.2 Contributions

We take a close look at the encryption schemes used in Kerberos v.5 (according to its specifications [26, 25]), in order to prove them secure, in the IND-CCA and INT-CTXT sense, assuming the underlying building blocks (e.g. a blockcipher) are secure. Our results complement the formal methods-based analysis of Kerberos as a key establishment and authentication protocol [17, 18].

General Profile. We first look at the encryption scheme description in the current version, v.5, specification (cf. [26], Section 6). We will refer to it as "General Profile". Fix a blockcipher with input-output length n, and a key for it; a checksum, i.e. a hash function with arbitrary input length, and output length l. A message M is first padded to make the length of the message plus l a multiple of n. Next, a random n-bit string *conf* is chosen. Then the checksum is applied to the string $\mathit{conf} \,\|\, 0^l \,\|\, M$. Let us call the checksum's output σ. Finally, the blockcipher in the CBC mode with fixed initial vector $IV = 0^n$ is applied to the string $\mathit{conf} \,\|\, \sigma \,\|\, M$. Decryption is defined accordingly. Security of the scheme depends on how the checksum function is instantiated. The

[1] We will often refer to encryption schemes whose goal is to provide both privacy and authenticity as *authenticated* encryption.

suggested instantiation is a hash function. We observe that the General Profile scheme conforms to a general Encode-then-Checksum-then-Encrypt construction, and the latter has a weakness. Namely, we show that even if one assumes the "more secure" component options, e.g. a secure blockcipher in a secure encryption mode and a secure hash function, the Encode-then-Checksum-then-Encrypt construction is not secure in general. That is, there exist attacks on the scheme composed of certain secure components, which shows that it does not provide integrity of ciphertexts. We note that these attacks do not apply to the General Profile itself, as it uses a particular encryption scheme recommended in [26]. Nevertheless, the attacks show a weakness in the overall design.

Modified General Profile. We propose simple, easy to implement modifications that are sufficient for provable security of the design of General Profile. Namely, we show that if the scheme uses a secure blockcipher (a pseudorandom function, or "PRF") in the CBC mode, as specified by General Profile, and if a message authentication code (MAC) that is a PRF is used as a checksum in place of the hash function, then Modified General Profile yields an encryption scheme that is IND-CCA and INT-CTXT secure. In particular, AES that is assumed to be a PRF and HMAC [8] that is proven to be a PRF [7], assuming the underlying compression hash function is a PRF, are good candidates for a blockcipher and MAC, respectively.

Simplified Profile. Next, we look at the recently proposed revisions to the encryption design in Kerberos, known as Simplified Profile (cf. Section 5 in [26] and [25]). This encryption scheme, for which implementations have not caught up yet, recommends to use AES or Triple-DES as a blockcipher, and HMAC [8] as a MAC, in the following manner. The message is first encoded such that the necessary padding is appended, and a random confounder (the name was suggested in most Kerberos specifications) is prepended. The blockcipher in CBC mode or a variant of CBC mode with ciphertext-stealing[2], both with fixed all-zero-bit IV, and HMAC are applied to the encoded message independently to yield two parts of the resulting ciphertext. Decryption is defined accordingly. We prove that this method yields an encryption scheme that is IND-CCA and INT-CTXT secure, under the assumption that the blockcipher and the MAC are PRF. This confirms soundness of the design of the Simplified Profile. AES is conjectured to be a PRF, Triple DES was shown to be a PRF in the ideal cipher model [15], and as mentioned before, HMAC was proven to be a PRF [7], assuming the underlying compression hash function is a PRF. Therefore, they are the right choices of instantiations for the Simplified Profile. We comment that even though the Simplified Profile uses the CBC scheme with a fixed IV, this does not compromise security because pre-pending a random confounder to the message before encrypting makes the scheme equivalent to the CBC with random IV.

While our results are not as unexpected or "catchy" as some results discovering a flaw or implementing an attack on a practical protocol, they are far from being less important. Having provable security guarantees is an invaluable benefit for any cryptographic design, especially a widely deployed protocol. Our results together with the formal methods-based results in the symbolic setting constitute strong provable security support for the design of Kerberos.

1.3 Related work

Bellare and Namprempre [12] study various ways to securely compose secure (IND-CPA) encryption and secure (unforgeable against chosen-message attacks, or "UF-CMA") message authentication

[2]Even though we analyze Simplified Profile only with "plain" CBC mode of encryption, we note that our analysis can easily be extended to CBC mode with ciphertext-stealing, and the results remain unaltered.

code schemes. They show that only one out of the three most straightforward composition methods, Encrypt-then-MAC, is secure in general (i.e. always yields an IND-CCA and INT-CTXT encryption scheme). At the same time, certain secure components can yield a scheme, constructed via Encrypt-and-MAC or MAC-then-Encrypt paradigms, that is not IND-CCA or not INT-CTXT. If Kerberos' design had utilized the Encrypt-then-MAC composition method with secure encryption and MAC schemes, we would have nothing to prove here. However, Kerberos uses some variations of Encrypt-and-MAC and MAC-then-Encrypt methods that rely on the properties of the encodings of the message, i.e. of preprocessing of the message before encryption and MAC are applied.

Bellare et al. [11] analyze security of encryption in another widely deployed protocol, Secure Shell, also known as SSH. They suggest several modifications to the SSH encryption to fix certain flaws, and they prove that the resulting scheme provably provides privacy against chosen-ciphertext attacks, and integrity of ciphertexts. They also provide general results about security of stateful encryption schemes composed according to the Encode-then-Encrypt-and-MAC paradigm, assuming certain security properties of the base encoding, encryption, and MAC schemes. The encryption scheme proposed for the revision of Kerberos v.5 (cf. Simplified Profile in [26]) conforms to the Encode-then-Encrypt-and-MAC method. However, the security results from [11] do not directly imply strong security notions of the Simplified Profile in Kerberos. First, the general results from [11] do not guarantee a strong notion of integrity of ciphertexts; they only consider a weaker notion of integrity of plaintexts. Second, the result of [11] require an IND-CPA secure base encryption scheme, but as we mentioned above, the base encryption in Kerberos is CBC with fixed IV, which is not IND-CPA secure.

Krawczyk [22] shows that the MAC-then-Encrypt composition method yields a secure authenticated encryption scheme, if the underlying MAC is UF-CMA, and if the encryption scheme uses a PRF blockcipher in CBC with random IV mode. We cannot use this result to prove security of the Modified General Profile, because the latter uses the CBC mode with zero IV, and moreover, it uses a particular encoding scheme so that the confounder (that basically plays the role of random IV for CBC) is also being MACed and encrypted. Proving this scheme requires special care.

Accordingly, we need to analyze the authenticated encryption schemes in Kerberos from scratch.

1.4 Outline

After defining some notation, we recall the relevant cryptographic primitives and their security definitions. Next, we outline the designs of schemes in the General and Simplified Profile authenticated encryption schemes of Kerberos' specification, as well as the modification to the General Profile that we propose. We follow with a detailed security analysis of the schemes, and we conclude with the summary.

2 Preliminaries

2.1 Notation

We denote by $\{0,1\}^*$ the set of all binary strings of finite length. If X is a string, then $|X|$ denotes its length in bits. If X, Y are strings, then $X \parallel Y$ denotes the concatenation of X and Y. For an integer k and a bit b, b^k denotes the string consisting of k consecutive "b" bits. For a string X, whose length is a multiple of integer n bits, $X[i]$ denotes the i^{th} block, meaning $X = X[1]||\ldots||X[l]$, where $l = |X|/n$, and $|X[i]| = n$, for all $i = 1, \ldots, l$. For any integers $0 \leq i < j \leq |M|$, $M_{[i]}$ denotes

the i^{th} bit of M, and $M_{[i...j]}$ denotes $M_{[i]} \| \ldots \| M_{[j]}$. If S is a set, then $|S|$ denotes the size of S; $X \xleftarrow{\$} S$ denotes that X is selected uniformly at random from S. If A is a randomized algorithm, then the notation $X \xleftarrow{\$} A$ denotes that X is assigned the outcome of the experiment of running A. If A is deterministic, we drop the dollar sign above the arrow.

2.2 Cryptographic Primitives and their Security

Symmetric Encryption.

Definition 2.1. [Symmetric encryption scheme] A *symmetric encryption scheme* $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, with associated message space MsgSp, is defined by three algorithms:

- The randomized *key generation* algorithm $\mathcal{K}$ returns a secret key K.
- The (possibly) randomized or stateful *encryption* algorithm $\mathcal{E}$ takes as input the secret key K and a plaintext $M \in$ MsgSp, and returns a ciphertext.
- The deterministic *decryption* algorithm $\mathcal{D}$ takes the secret key K and a ciphertext C to return the corresponding plaintext, or the special symbol $\perp$ indicating that the ciphertext was invalid.

The consistency condition requires that $\mathcal{D}_K(\mathcal{E}_K(M)) = M$, for all K that can be output by $\mathcal{K}$, and all $M \in$ MsgSp.

We now recall cryptographic security notions for encryption. The following definition [9] is for data privacy (confidentiality). It formalizes the requirement that even though an adversary may know some partial information about the data, no additional information is leaked.

Definition 2.2. [IND-CPA, IND-CCA] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme. For $\text{atk} \in \{\text{cpa}, \text{cca}\}$, adversary A, and a bit b, define the experiments $\mathbf{Exp}^{\text{ind-atk-b}}_{\mathcal{SE}}(A)$ as follows. In all the experiments, first the key K is generated by $\mathcal{K}$. Let LR (left-or-right) be the "selector" that on input M_0, M_1, b returns M_b. The adversary A is given access to the *left-right encryption oracle* $\mathcal{E}_K(\text{LR}(\cdot, \cdot, b))$ that it can query on any pair of messages of equal length in MsgSp. In $\mathbf{Exp}^{\text{ind-cca-b}}_{\mathcal{SE}}(A)$, the adversary is also given the decryption oracle $\mathcal{D}_K(\cdot)$ that it can query on any ciphertext that was not returned by the other oracle. The adversary's goal is to output a bit d as its guess of the challenge bit b; the experiment returns d as well. The *ind-atk-advantage* of an adversary A is defined as:

$$\mathbf{Adv}^{\text{ind-atk}}_{\mathcal{SE}}(A) \quad = \quad \Pr\left[\, \mathbf{Exp}^{\text{ind-atk-1}}_{\mathcal{SE}}(A) = 1 \,\right] - \Pr\left[\, \mathbf{Exp}^{\text{ind-atk-0}}_{\mathcal{SE}}(A) = 1 \,\right].$$

The scheme $\mathcal{SE}$ is said to be *indistinguishable against chosen-plaintext attack* or *IND-CPA* (resp., *chosen-ciphertext attack* or *IND-CCA*), if for every adversary A with reasonable resources, its ind-cpa (resp., ind-cca) advantage is small[3].

[3]The resources of an adversary we care about are its running time, the number of oracle queries it makes, and the total length of the queries. We use the standard convention that running time of an adversary is measured with respect to the entire experiment in which it runs. Here, and further in the paper, we call the resources of an algorithm (or adversary) "reasonable" if it runs for some reasonable amount of time (e.g. up to 10 years, or does 2^{60} basic operations in some fixed model of computation) and does a reasonable number of oracle queries of reasonable length. We call the value of an advantage "small" if it is very close to 0 (e.g. 2^{-20}.) In general, "reasonable" parameters depend on a particular application. In computing the total length of queries made to the LR encryption oracle, we only count the length of one of the messages, instead of the total length of the message pair (M_0, M_1).

It is easy to see that IND-CCA security is a stronger notion that implies IND-CPA security.

The following definition [12, 13] is for authenticity and integrity of encryption. It formalizes the requirement that no adversary should be able to compute a new ciphertext which the receiver will deem valid.

Definition 2.3. [INT-CTXT] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme. The encryption scheme is said to provide *authenticity, or ciphertext integrity (be INT-CTXT secure)*, if any adversary A with reasonable resources can be successful in the following experiment only with small probability, called the *int-ctxt-advantage* of A, $\mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(A)$. In the experiment, first the key K is generated by $\mathcal{K}$. The adversary has access to two oracles: encryption oracle $\mathcal{E}_K(\cdot)$ and verification oracle $\mathcal{V}_K(\cdot)$. On input, a ciphertext C, $\mathcal{V}_K(\cdot)$ returns 1 if C was not returned by $\mathcal{E}_K(\cdot)$ and $\mathcal{D}_K(C) \neq \perp$. The adversary is successful in the experiment if $\mathcal{V}_K(\cdot)$ ever returns 1.

It has been shown [12] that if an encryption scheme is IND-CPA and INT-CTXT, then it is also IND-CCA.

Theorem 2.4. [**[12], Theorem 3.2**] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme. If it is IND-CPA and INT-CTXT secure, then it is also IND-CCA secure. Concretely, for any adversary A attacking the IND-CCA security of $\mathcal{SE}$, that runs in time t, and makes q_e queries to the left-right encryption oracle, and q_d queries to the decryption oracle, totaling μ_e and μ_d bits, respectively, there exist adversaries B and C attacking the scheme's IND-CPA and INT-CTXT security, respectively, such that

$$\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}}(A) \;\leq\; \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(B) + 2 \cdot \mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(C)\,.$$

Furthermore, B runs in time[4] t, and makes q_e queries to the left-right encryption oracle, totaling μ_e bits, while C runs in time t, and makes q_e queries to the encryption oracle, and q_d queries to the verification oracle, totaling μ_e and μ_d bits, respectively.

Pseudorandom Function Families. A family of functions is a map $E\colon \mathit{Keys} \times \mathit{Dom} \to \mathit{Ran}$, where we regard *Keys* as the *keyspace* for the function family in that a *key* $K \in \mathit{Keys}$ induces a particular function from this family, which we denote by $E_K(\cdot)$.

Definition 2.5. [PRF] Let $E\colon \mathit{Keys} \times \mathit{Dom} \to \mathit{Ran}$ be a function family. Let R be the set of all functions from *Dom* to *Ran*. E is called *pseudorandom, or PRF secure*, if any adversary A, with reasonable resources and access to an oracle that it can query on messages in MsgSp, has small *prf-advantage* defined as

$$\mathbf{Adv}^{\text{prf}}_{E}(A) \;=\; \Pr\left[\, K \xleftarrow{\$} \mathit{Keys} \;:\; A^{E_K(\cdot)} = 1 \,\right] - \Pr\left[\, g \xleftarrow{\$} R \;:\; A^{g(\cdot)} = 1 \,\right]\,.$$

Message Authentication Codes (MACs).

Definition 2.6. [MAC] A *message authentication code* $\mathcal{MAC} = (\mathcal{K}, \mathcal{T})$ with associated *message space* MsgSp is defined by two algorithms:

- The randomized *key generation* algorithm $\mathcal{K}$ returns a secret key K.

[4]The time complexity given in [12] is slightly different due to the difference in convention.

- The deterministic[5] *tagging* algorithm $\mathcal{T}$ takes as input the secret key K, and a plaintext $M \in$ MsgSp to return a tag for M.

For a message-tag pair (M, σ), we say σ is a valid tag for M, if $\sigma = \sigma'$, where $\sigma' \leftarrow \mathcal{T}_K(M)$.

The following security definition [10] requires that any adversary with reasonable resources can forge a valid tag for a new message only with small probability.

Definition 2.7. [UF-CMA] Let $\mathcal{MAC} = (\mathcal{K}, \mathcal{T})$ be a MAC scheme. It is called *unforgeable against chosen-message attacks, or UF-CMA secure*, if any adversary A with reasonable resources can be successful in the following experiment, only with small probability, called the *uf-cma-advantage* of A, $\mathbf{Adv}^{\text{uf-cma}}_{\mathcal{MAC}}(A)$. In the experiment, first the random key K is generated by $\mathcal{K}$. The adversary has access to two oracles: tagging oracle $\mathcal{T}_K(\cdot)$ and verification oracle[6] $\mathcal{V}_K(\cdot,\cdot)$. On input, a message-tag pair (m, σ), $\mathcal{V}_K(\cdot,\cdot)$ returns 1 if m was not queried to $\mathcal{T}_K(\cdot)$ and $\mathcal{T}_K(m) = \sigma$, otherwise it returns 0. The adversary is successful in the experiment if $\mathcal{V}_K(\cdot,\cdot)$ ever returns 1.

Another (stronger) security definition requires that the output of the MAC is indistinguishable from a random string. The definition below is very similar to the PRF definition for the function family. The only difference is that now the key generation algorithm is used to generate a key.

Definition 2.8. [PRF for MACs] Let $\mathcal{MAC} = (\mathcal{K}, \mathcal{T})$ be a MAC scheme. Let R be the set of all functions with the same domain and range as $\mathcal{T}$. $\mathcal{MAC}$ is called *pseudorandom, or PRF secure*, if any adversary A with reasonable resources and access to an oracle that it can query on messages in MsgSp, has a small *prf-advantage* defined as

$$\mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(A) \;=\; \Pr\left[\, K \xleftarrow{\$} \mathcal{K} \;:\; A^{\mathcal{T}_K(\cdot)} = 1 \,\right] - \Pr\left[\, g \xleftarrow{\$} R \;:\; A^{g(\cdot)} = 1 \,\right] .$$

We recall the fact that any MAC that is PRF is also UF-CMA.

Theorem 2.9. [**[14], Proposition 6.3**] Let $\mathcal{MAC} = (\mathcal{K}, \mathcal{T})$ be a MAC scheme. Then, for any adversary F attacking UF-CMA security of $\mathcal{MAC}$, that runs in time t, and makes q_t queries to the tagging oracle, and q_v queries to the verification oracle, totaling μ_t and μ_v bits, respectively, there exists an adversary G attacking the PRF security of $\mathcal{MAC}$, such that

$$\mathbf{Adv}^{\text{uf-cma}}_{\mathcal{MAC}}(F) \;\leq\; \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G) + \frac{q_v}{|Ran_{\mathcal{T}}|} ,$$

where $Ran_{\mathcal{T}}$ denotes the range of $\mathcal{T}$. Furthermore, G runs in time t, and makes $(q_t + q_v)$ oracle queries, totaling $(\mu_t + \mu_v)$ bits.

HASH FUNCTIONS.

Definition 2.10. [Hash function] A *hash function* $HF = (\mathcal{K}, \mathcal{H})$ consists of two algorithms. The *key generation* algorithm $\mathcal{K}$ outputs a key K. The deterministic *hash* algorithm $\mathcal{H}$ on inputs K, and $M \in \{0,1\}^*$, outputs the hash value H.

[5] A MAC does not have to be deterministic, but most practical schemes are, and in this paper we consider only deterministic MACs.

[6] Since we only consider deterministic MACs, the verification oracle is not necessary. However, we keep it for generality. In computing the total length of the queries made to the verification oracle, we only count the length of message m, and not the message-tag pair (m, σ).

Definition 2.11. **[Collision-resistance]** A hash function $HF = (\mathcal{K}, \mathcal{H})$ is called *collision-resistant* if any adversary with reasonable resources that is given a random K output by $\mathcal{K}$ can output two messages $M_1, M_2 \in \{0,1\}^*$, such that $\mathcal{H}_K(M_1) = \mathcal{H}_K(M_2)$, and $M_1 \neq M_2$, only with a small probability.

ENCODING SCHEME. An encoding scheme is an unkeyed invertible transformation that is used to extend the message with some associated data, such as padding, a counter, random nonce, etc.

Definition 2.12. **[Encoding scheme]** An *encoding scheme* $\mathcal{EC} = (Encode, Decode)$ with associated *message space* MsgSp is defined by two algorithms. The (possibly) randomized or stateful *encoding* algorithm *Encode* takes a message $M \in$ MsgSp, and outputs a pair of messages (M_e, M_t). The deterministic *decoding* algorithm takes M_e and returns a pair (M, M_t), or $(\perp, \perp)$ on error.

For any message $M \in$ MsgSp, let $(M_e, M_t) \xleftarrow{\$} Encode(M)$, and $(M', M_t') \leftarrow Decode(M_e)$. Then the consistency condition requires that $M = M'$ and $M_t = M_t'$. We note that in the constructions we will consider, M_e is going to be used as input to the encryption algorithm, and M_t is going to be used as input to the MAC algorithm.

The following is from [13, 11].

Definition 2.13. **[Coll-CPA]** Let $\mathcal{EC} = (Encode, Decode)$ be an encoding scheme. It is called *collision-resistant against chosen-plaintext attacks, or Coll-CPA*, if any adversary A with reasonable resources has only small success probability, called the *coll-cpa-advantage* of A, or $\mathbf{Adv}^{\text{coll-cpa}}_{\mathcal{EC}}(A)$, in the following experiment. The adversary has access to the encoding oracle $Encode(\cdot)$, and it is considered successful if it ever gets two replies (M_e, M_t), (M_e', M_t'), such that $M_t = M_t'$.

3 Analysis of Encryption in Kerberos v.5

3.1 General Profile and the Underlying Composition Method

We first look at the encryption scheme specified in [26]. This document describes several options, but we note that all the choices conform to a general composition method that we outline below (the design is further generalized in [21]).

Construction 3.1. **[Encode-then-Checksum-then-Encrypt]** Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$, $\mathcal{EC} = (Encode, Decode)$, and $\mathcal{CS} = (\mathcal{K}_t, \mathcal{T})$ be an encryption scheme, an encoding scheme, and a checksum (i.e. hash function or MAC). The message space of the corresponding *Encode-then-Checksum-then-Encrypt* scheme $\mathcal{SE}' = (\mathcal{K}', \mathcal{E}', \mathcal{D}')$ is that of $\mathcal{EC}$, and the rest of the algorithms are defined as follows:

- $\mathcal{K}'$ runs $\mathcal{K}_e, \mathcal{K}_t$, and returns their outputs $K_e \parallel K_t$.
- $\mathcal{E}'$ on inputs $K_e \parallel K_t$ and M, first gets the encodings via $(M_e, M_t) \xleftarrow{\$} Encode(M)$. It then computes $\sigma \leftarrow \mathcal{T}_{K_t}(M_t)$, parses M_e as $M_{el} \| M_{er}$, and returns $C \xleftarrow{\$} \mathcal{E}_{K_e}(M_{el} \parallel \sigma \parallel M_{er})$.
- $\mathcal{D}'$ on inputs $K_e \parallel K_t$ and C, computes $M_e \leftarrow M_{el} \parallel M_{er}$, σ from $(M_{el} \parallel \sigma \parallel M_{er}) \leftarrow \mathcal{D}_{K_e}(C)$, decodes $(M, M_t) \leftarrow Decode(M_e)$, computes $\sigma' \leftarrow \mathcal{T}_{K_t}(M_t)$, and returns M, if $\sigma = \sigma'$, and $\perp$ otherwise.

Above, we assume that the outputs of the encoding scheme are compatible with the inputs to $\mathcal{E}, \mathcal{T}$. Figure 1 illustrates the design.

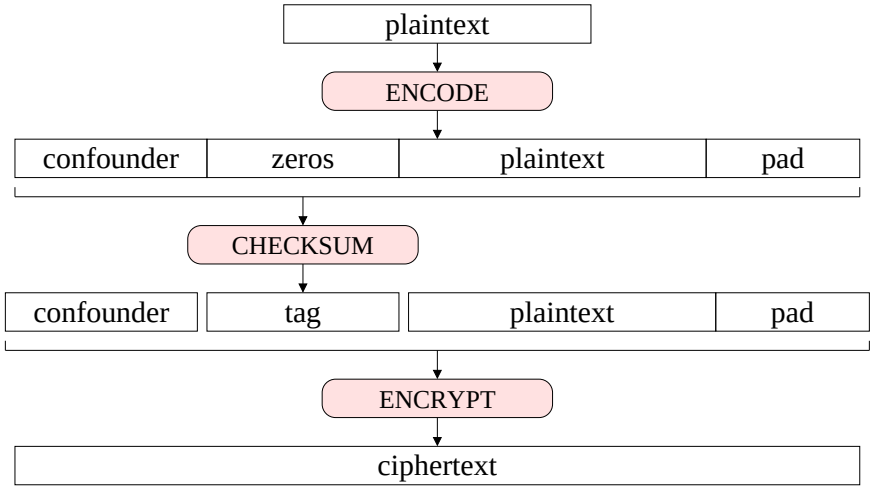

Figure 1: Encode-then-Checksum-then-Encrypt paradigm used in General Profile of Kerberos v.5.

The next construction specifies in more detail how Kerberos' encryption operates, i.e. what specific algorithms instantiate the generic composition method of Construction 3.1.

Construction 3.2. [Authenticated encryption in Kerberos: General Profile] Let $E\colon \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a blockcipher. Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be the associated CBC encryption mode (cf. [9] for the formal description), with $IV = 0^n$ [7]. Let $HF = (\mathcal{K}_h, \mathcal{H})$ be a hash function (to be used as checksum $\mathcal{CS}$) with output of length l, which is keyless or whose key is public. Let $\mathcal{EC} = (Encode, Decode)$ be an encoding scheme, such that $Encode$ with MsgSp $= \{0,1\}^*$, on input M, pads it to make the length of $(l + |M|)$ a multiple of n bits (so that decoding is unambiguous), picks a random confounder of n bits $conf \xleftarrow{\$} \{0,1\}^n$, computes $M_e \leftarrow conf \parallel M$, and $M_t \leftarrow conf \parallel 0^l \parallel M$, and returns (M_e, M_t). M_{el} is defined to be $conf$, and M_{er} is the rest of M_e. $Decode$ on input M_e parses it as $conf \| M$, computes $M_t \leftarrow conf \| 0^l \| M$, and returns (M, M_t). Then Construction 3.1 describes the authenticated encryption called General Profile[8].

Security Analysis of General Profile. Some supported instantiations include DES as the blockcipher, and MD4 and MD5 as the hash function. These are not good choices for known reasons. DES is an outdated standard, since its key and block sizes are too small considering modern computing power, and collisions have been found in MD4 and MD5 [28]. However, what our results show is that using the "more secure" building blocks, such as AES and a collision-resistant hash function, will not necessarily solve the problem. More precisely, we can neither prove, nor disprove the security of the General profile in this case. What we can show is that the Encode-then-Checksum-then-Encrypt composition method does not provide integrity in general

[7] The Kerberos specification also allows stateful update of the IV, i.e. the IV is assigned to be the last block of the previous ciphertext. Our analyses apply to this case as well. But, since this option is not commonly used, we do not consider it in detail. We note however, that [26] does not specify how the state and IV are updated when the receiver gets an invalid ciphertext. The only reasonable resolution preventing malicious attacks disrupting the future communication may be to issue an error message and reset the IV to 0^n.

[8] Our analysis does not take into account stateful approaches for key derivation used in few options of General profile.

when it uses a hash function as a checksum, even if it uses a secure encryption option for the underlying encryption scheme. We note that the theorem below does not imply that the General Profile (Construction 3.2) is insecure, but it shows the limitation of its underlying general design (Construction 3.1), when used with the given encoding scheme.

Theorem 3.3. Let $\mathcal{EC} = (Encode, Decode)$ be the encoding scheme, defined in Construction 3.2. There exists an IND-CPA secure encryption scheme, and a collision-resistant hash function, so that the authenticated encryption obtained via Encode-then-Checksum-then-Encrypt (Construction 3.1), does not provide integrity (is not INT-CTXT secure). Concretely, there exists an adversary I with reasonable resources, such that $\mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}'}(I) = 1$.

The proof is in Section 4.1. In fact, the proof also shows that the general construction is insecure, even when a secure MAC is used as checksum (with the corresponding secret key being secret, of course), but in this case, the attack makes use of a rather artificial IND-CPA scheme. The attacks we provide are similar to those in [12, 22] that show insecurity of several general composition methods. We repeat that the attacks that we provide in the proof do not translate into an attack on any of the recommended options, they just show limitations in the general composition method.

Modified General Profile. We now suggest simple and easy-to-implement modifications to the General profile construction, and show that they are sufficient to prove the security of the scheme. Namely, we suggest to use a secure MAC in place of the hash function, and show that the resulting authenticated encryption scheme is secure. Note that this does not contradict Theorem 3.3, because now we look at the particular encryption scheme that the General Profile uses (Construction 3.2), i.e. CBC with zero IV. We now define the construction, and state its security.

Construction 3.4. **[Modified General profile]** The construction is like Construction 3.2, except that a message authentication code $\mathcal{MAC} = (\mathcal{K}_t, \mathcal{T})$ is used as checksum $\mathcal{CS}$.

Theorem 3.5. The authenticated encryption scheme described by the Modified General Profile (Construction 3.4) is INT-CTXT and IND-CCA secure, if the underlying blockcipher is a PRF, and the underlying checksum (MAC) is a PRF.
Concretely, let $E\colon \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a blockcipher, and $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be the CBC encryption mode with $IV = 0^n$, that uses E. Let $\mathcal{MAC} = (\mathcal{K}_t, \mathcal{T})$ be a message authentication code with output of length l. Let $\mathcal{EC} = (Encode, Decode)$ be an encoding scheme, and $\mathcal{SE}' = (\mathcal{K}', \mathcal{E}', \mathcal{D}')$ be the authenticated encryption scheme associated to them by Modified General Profile (Construction 3.4). Then, for any adversary I attacking the INT-CTXT security of $\mathcal{SE}'$, that runs in time t, and makes q'_e queries to the encryption oracle, and q_v queries to the verification oracle, totaling μ'_e and μ_v bits, respectively, there exists an adversary F attacking the UF-CMA security of $\mathcal{MAC}$, such that

$$\mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}'}(I) \;\leq\; \mathbf{Adv}^{\text{uf-cma}}_{\mathcal{MAC}}(F)\,. \tag{1}$$

Furthermore, F runs in time t, and makes q'_e queries to the tagging oracle,and q_v queries to the verification oracle, totaling at most $\mu'_e + q'_e \cdot (2n + l - 1)$ and $\mu_v + q_v \cdot (2n + l - 1)$ bits, respectively. And for any adversary A attacking the IND-CCA security of $\mathcal{SE}'$, that runs in time t, and makes q_e queries to the left-right encryption oracle and q_d queries to the decryption oracle, totaling μ_e and μ_d bits, respectively, there exist adversaries B and G attacking PRF security of E and $\mathcal{MAC}$, respectively, such that

$$\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}'}(A) \;\leq\; \mathbf{Adv}^{\text{prf}}_{E}(B) + 4 \cdot \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G) + \frac{\mu_e^2}{n^2 \cdot 2^n} + \frac{2 \cdot q_d}{2^l}\,. \tag{2}$$

Furthermore, B runs in time t, and makes at most $\lfloor(\mu_e + q_e \cdot (2n + l - 1)/n)\rfloor$ oracle queries, totaling at most $\mu_e + q_e \cdot (2n + l - 1)$ bits; G runs in time t, and makes $q_e + q_d$ oracle queries, totaling at most $\mu_e + \mu_d + (q_e + q_d)(2n + l - 1)$ bits.

The proof is in Section 4.2. Note that the INT-CTXT security of the scheme requires only UF-CMA security of the checksum (MAC), while IND-CCA security relies on it being a PRF. As we mentioned before, any PRF MAC is also UF-CMA (Theorem 2.9), so PRF security is a sufficient assumption.

AES is believed to be a PRF, and HMAC was shown to be a PRF [7], assuming the underlying compression function is a PRF (cf. [7] for the definition of the latter). Therefore, these schemes constitute good instantiations for the above design.

3.2 Simplified Profile and the Underlying Composition Method

Kerberos designers proposed a new construction that they call "Simplified profile" (cf. Section 5 in [26], and [25]). Again, we start with a more general composition method that outlines the design.

Construction 3.6. [Encode-then-Encrypt&MAC] Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$, $\mathcal{MAC} = (\mathcal{K}_t, \mathcal{T})$, and $\mathcal{EC} = (Encode, Decode)$ be an encryption scheme, a MAC scheme, and an encoding scheme. The message space of corresponding *Encode-then-Encrypt&MAC* scheme $\mathcal{SE}' = (\mathcal{K}', \mathcal{E}', \mathcal{D}')$, is that of $\mathcal{EC}$, and the algorithms are defined as follows:

- $\mathcal{K}'$ runs $\mathcal{K}_e, \mathcal{K}_t$, and returns their outputs $K_e \parallel K_t$.
- $\mathcal{E}'$ on inputs $K_e \parallel K_t$ and M, first gets the encodings via $(M_e, M_t) \xleftarrow{\$} Encode(M)$. It then computes $C \leftarrow \mathcal{E}_{K_e}(M_e)$, $\sigma \leftarrow \mathcal{T}_{K_t}(M_t)$, and returns $C \parallel \sigma$.
- $\mathcal{D}'$ on inputs $K_e \parallel K_t$ and $C \parallel \sigma$, computes $M_e \leftarrow \mathcal{D}_{K_e}(C)$, decodes $(M, M_t) \leftarrow Decode(M_e)$, computes $\sigma' \leftarrow \mathcal{T}_{K_t}(M_t)$, and returns M, if $\sigma = \sigma'$, and $\perp$ otherwise.

Above we assume that the outputs of the encoding scheme are compatible with the inputs to $\mathcal{E}, \mathcal{T}$.

The next construction defines the Simplified profile, and Figure 2 depicts the design.

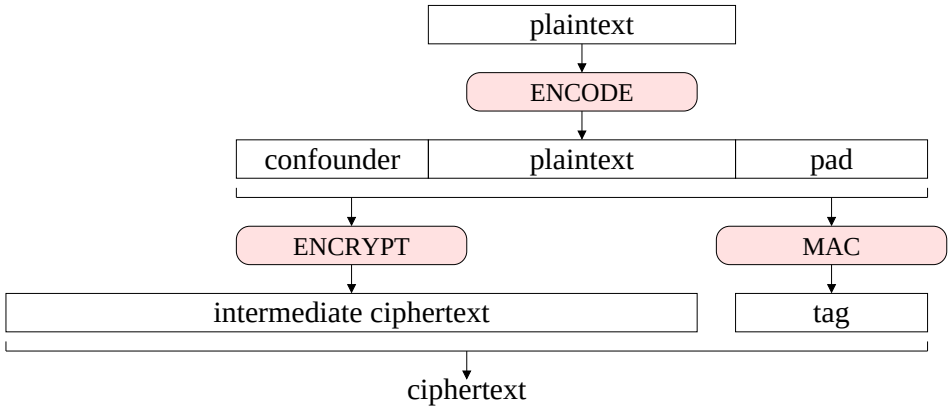

Figure 2: Encode-then-Encrypt&MAC paradigm used in Simplified profile of Kerberos v.5.

Construction 3.7. [Authenticated encryption in Kerberos: Simplified profile] Let $E\colon \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a blockcipher. Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be the associated CBC encryption mode with $IV = 0^n$. Let $\mathcal{MAC} = (\mathcal{K}_t, \mathcal{T})$ be a MAC scheme. Let $\mathcal{EC} = (Encode, Decode)$ be an encoding scheme, such that $Encode$ with $\mathrm{MsgSp} = \{0,1\}^*$ on input M pads M to make its length a multiple of n bits (while permitting unambiguous decoding), picks a random confounder of n bits $conf \xleftarrow{\$} \{0,1\}^n$, computes $M_e \leftarrow conf \,\|\, M$, and $M_t \leftarrow conf \,\|\, M$, and returns (M_e, M_t). $Decode$ on input M_e, parses it as $conf \| M$, computes $M_t \leftarrow M_e$, and returns (M, M_t). Then Construction 3.6 describes the Simplified Profile of authenticated encryption in Kerberos.

The following theorem states that the Simplified Profile provides strong security guarantees.

Theorem 3.8. The authenticated encryption scheme $\mathcal{SE}'$, described by the Simplified Profile (Construction 3.7), is INT-CTXT and IND-CCA secure if the underlying blockcipher E is a PRF and the underlying MAC is a PRF.
Concretely, let $E\colon \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a blockcipher, and $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be the CBC encryption mode with $IV = 0^n$ that uses E. Let $\mathcal{MAC} = (\mathcal{K}_m, \mathcal{T})$ be a MAC scheme and $\mathcal{EC} = (Encode, Decode)$ be an encoding scheme. Let $\mathcal{SE}'$ be the authenticated encryption scheme associated to them by Simplified Profile (Construction 3.7). Then, for any adversary I attacking INT-CTXT security of $\mathcal{SE}'$, that runs in time t, and makes q'_e queries to the encryption oracle, and q_v queries to the verification oracle, totaling μ'_e and μ_v bits, respectively, there exists an adversary F attacking UF-CMA security of $\mathcal{MAC}$, such that

$$\mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}'}(I) \;\leq\; \mathbf{Adv}^{\text{uf-cma}}_{\mathcal{MAC}}(F)\,. \tag{3}$$

Furthermore, F runs in time t and makes q'_e queries to the tagging oracle and q_v queries to the verification oracle, totaling at most $\mu'_e + q'_e \cdot (2n-1)$ and $\mu_v + q_v \cdot (2n-1)$ bits, respectively.
And for any adversary A attacking IND-CCA security of $\mathcal{SE}'$, that runs in time t and makes q_e queries to the left-right encryption oracle, and q_d queries to the decryption oracle, totaling μ_e and μ_d bits, respectively, there exist adversaries B and G attacking PRF security of E and $\mathcal{MAC}$, respectively, such that

$$\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}'}(A) \;\leq\; \mathbf{Adv}^{\text{prf}}_{E}(B) + 4 \cdot \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G) + \frac{q_e(q_e-1)}{2^{n+1}} + \frac{\mu_e^2}{n^2 \cdot 2^n} + \frac{2 \cdot q_d}{|Ran_{\mathcal{T}}|}\,, \tag{4}$$

where $Ran_{\mathcal{T}}$ denotes the set of outputs of $\mathcal{T}$. Furthermore, B runs in time t and makes at most $\lfloor (\mu_e + q_e \cdot (2n-1))/n) \rfloor$ oracle queries, totaling at most $\mu_e + q_e \cdot (2n-1)$ bits; G runs in time t and makes $q_e + q_d$ oracle queries, totaling at most $\mu_e + \mu_d + (q_e + q_d) \cdot (2n-1)$ bits.

The proof is in Section 4.3. Note that INT-CTXT security of the scheme requires only UF-CMA security of the MAC, while IND-CCA security relies on the MAC being a PRF. As we mentioned before, any PRF MAC is also UF-CMA (Theorem 2.9), so PRF security is a sufficient assumption. Also, AES is believed to be a PRF, and HMAC was shown to be a PRF [7], assuming the underlying compression function is a PRF (cf. [7] for the definition of the latter notion). Therefore, these schemes constitute good instantiations for the above design.

4 Proofs

4.1 Proof of Theorem 3.3

We present two alternative proofs of insecurity of the Encode-then-Checksum-then-Encrypt paradigm used in General Profile authenticated encryption. The first proof uses a natural encryption scheme and a not-so-natural hash function as a checksum. The second proof uses a special encryption scheme, but the checksum can be instantiated with arbitrary secure MAC.

PROOF 1. Before presenting the proof, we give a high level idea. We show that the general authenticated encryption paradigm underlying General Profile does not preserve integrity of ciphertexts when instantiated with stateful counter (CTR) mode of the encryption scheme and a collision-resistant hash function that happens to leak the first bit of its input. The CTR mode of encryption is somewhat similar to the one-time pad, where the underlying blockcipher is applied to a counter to generate a pseudorandom pad which is then XORed with the message. Now, the ingenuity of our attack lies in showing that given any ciphertext that was output by the above scheme, one can produce another valid ciphertext by simply flipping the bits at two different positions, namely the first bit of the first and second blocks of the ciphertext. We repeat that the attack does not apply to the General Profile scheme itself.

Let $E\colon\ \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a blockcipher, and let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be the associated stateful counter encryption scheme, known as CTR or XOR encryption mode (cf. [9]). Its key generation algorithm $\mathcal{K}_e$ simply returns a random k-bit string K_e. The encryption algorithm $\mathcal{E}$ is stateful and maintains a counter ctr that is initially 0. $\mathcal{E}$ takes the key K_e, the current counter ctr, and a message M (padded if necessary to a length multiple of n-bits), and outputs $ctr \parallel C[1] \parallel C[2] \parallel ... \parallel C[m]$, where m is the total number of blocks, and for $1 \le i \le m$, $C[i] \leftarrow M_i \oplus E_{K_e}(\langle ctr + i \rangle)$. Here $\langle i \rangle$ denotes the n-bit representation of an integer i. Next, $\mathcal{E}$ updates the counter to $ctr + m + 1$. The decryption algorithm $\mathcal{D}$ takes K_e and a ciphertext $ctr \parallel C[1] \parallel ... \parallel C[m]$ and outputs $M[1] \parallel ... \parallel M[m]$, where for $1 \le i \le m$, $M[i] \leftarrow C[i] \oplus E_{K_e}(\langle ctr + i \rangle)$. The CTR encryption mode is proven to be IND-CPA secure if E is a PRF [9].

Let $HF = (\mathcal{K}_h, \mathcal{H})$ be a collision-resistant hash function whose hash algorithm outputs strings of length l. Consider a modified hash function $HF' = (\mathcal{K}_h, \mathcal{H}')$ whose hash algorithm $\mathcal{H}'$ on an input key K_h and a message M, outputs $M_{[0]} \parallel \mathcal{H}_{K_h}\left(M_{[1...|M|-1]}\right)$. We show that HF' is also collision-resistant.

For any adversary A that can find collisions in HF', we construct an adversary B that can find collisions in HF. B gives its own challenge key K_h to A and gets back two messages M', N'. B computes $M \leftarrow M'_{[1...|M'|-1]}$, $N \leftarrow N'_{[1...|N'|-1]}$, and outputs M, N. Note that $\mathcal{H}'_{K_h}(M') = M'_{[0]} \| \mathcal{H}_{K_h}\left(M'_{[1...|M'|-1]}\right) = M'_{[0]} \| \mathcal{H}_{K_h}(M)$, and $\mathcal{H}'_{K_h}(N') = N'_{[0]} \| \mathcal{H}_{K_h}\left(N'_{[1...|N'|-1]}\right) = N'_{[0]} \| \mathcal{H}_{K_h}(N)$. If $\mathcal{H}'_{K_h}(M') = \mathcal{H}'_{K_h}(N')$, and $M' \neq N'$, then it is easy to see that $\mathcal{H}_{K_h}(M) = \mathcal{H}_{K_h}(N)$, and $M \neq N$. B is almost as efficient as A. Therefore, if HF is collision resistant, then so is HF'.

We now present an adversary I that breaks the INT-CTXT security of the scheme described by Construction 3.1 when it uses CTR encryption mode and modified hash function HF' as $\mathcal{SE}$ and $\mathcal{CS}$, respectively. I selects an arbitrary n-bit-long message M and queries it to the encryption oracle. Let $ctr \| C$ be the oracle's reply. I then queries the ciphertext $ctr \| C'$ to the verification oracle, where C' is computed from C by flipping the first bit of the first and second blocks.

We claim that the int-ctxt advantage of I is 1. This is justified as follows. Consider $conf \| \sigma \| M = \mathcal{D}_{K_e}(ctr \| C)$. Here, $\sigma = \mathcal{H}'_{K_h}(M_t)$, and $M_t = conf \| 0^{l+1} \| M$. So $\sigma = conf_{[0]} \| \mathcal{H}_{K_h}\left(M_{t[1...|M_t|-1]}\right)$.

$ctr\|C$ can be parsed as $ctr\|C[1]\|C[2]\|D$, where $C[1]$ and $C[2]$ are the first and second blocks of C, and D is the remaining part of C. From the description of CTR encryption mode it is clear that

$C[1] = conf \oplus E_{K_e}(\langle ctr + 1\rangle)$, and

$C[2] = \big(\sigma\|M_{[0\ldots n-l-2]}\big) \oplus E_{K_e}(\langle ctr+2\rangle)$, where $\sigma = conf_{[0]}\|\mathcal{H}_{K_h}\big(M_{t[1\ldots|M_t|-1]}\big)$, and $M_{[0\ldots n-l-2]}$ is the first $n-(l+1)$ bits of M.

So $C[2] = \big(conf_{[0]}\|\mathcal{H}_{K_h}\big(M_{t[1\ldots|M_t|-1]}\big)\|M_{[0\ldots n-l-2]}\big) \oplus E_{K_e}(\langle ctr + 2\rangle)$.

Let us denote the ciphertext blocks produced by flipping the first bit of $C[1]$ and $C[2]$ by $C'[1]$ and $C'[2]$, respectively. So we have

$C'[1] = \big(\overline{conf_{[0]}}\|conf_{[1\ldots n-1]}\big) \oplus E_{K_e}(\langle ctr + 1\rangle)$ and

$C'[2] = \big(\overline{conf_{[0]}}\|\mathcal{H}_{K_h}\big(M_{t[1\ldots|M_t|-1]}\big)\|M_{[0\ldots n-l-2]}\big) \oplus E_{K_e}(\langle ctr + 2\rangle)$.

Let us denote the decryption of $ctr\|C'$ by $(M'_{el}\|\sigma'\|M'_{er})$. So we have $M'_{el} = (\overline{conf_{[0]}}\|conf_{[1\ldots n-1]})$, $\sigma' = (\overline{conf_{[0]}}\|\mathcal{H}_{K_h}(M_{t[1\ldots|M_t|-1]}))$, and $M'_{er} = M$.

Now notice that

$M'_e = (M'_{el}\|M'_{er}) = (\overline{conf_{[0]}}\|conf_{[1\ldots n-1]})\|M$, and

$M'_t = (M'_{el}\|0^{l+1}\|M'_{er}) = (\overline{conf_{[0]}}\|conf_{[1\ldots n-1]})\|0^{l+1}\|M$.

It is clear $M_{t[1\ldots|M_t|-1]} = M'_{t[1\ldots|M'_t|-1]}$ because M_t and M'_t differ only in first bit. So from above, we have

$\sigma' = \overline{conf_{[0]}}\|\mathcal{H}_{K_h}(M_{t[1\ldots|M_t|-1]}) = \overline{conf_{[0]}}\|\mathcal{H}_{K_h}(M'_{t[1\ldots|M'_t|-1]}) = \mathcal{H}'_{K_h}(M'_t)$.

Thus, (M'_t, σ') is a valid message-tag pair. Hence $ctr\|C'$ is a valid ciphertext that was never returned by the encryption oracle, and therefore, the int-ctxt advantage of I is 1.

I makes one oracle query of length n bits, and performs two operations of bit-complementation. □

Proof 2. Our second proof is relatively simpler than the first one. We show that the general authenticated encryption paradigm underlying General Profile does not preserve integrity of ciphertexts when instantiated with any arbitrary encoding scheme, an unforgeable MAC, and a special type of IND-CPA secure encryption scheme whose encryption algorithm prepends zero to the ciphertext and the decryption algorithm simply ignores the first bit of the ciphertext. In the Encode-then-Checksum-then-Encrypt paradigm encryption is the last step. So, with the above mentioned special type of encryption, one can easily produce a new valid ciphertext C', given any ciphertext C output by the above scheme by flipping the first bit of C. We repeat that the attack does not apply to the General Profile scheme itself.

Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be any IND-CPA secure encryption scheme. Consider a modified encryption scheme $\mathcal{SE}'' = (\mathcal{K}_e, \mathcal{E}'', \mathcal{D}'')$, where $\mathcal{E}''$ on input key K_e and a message M outputs $0\|\mathcal{E}_{K_e}(M)$, and $\mathcal{D}''$ on input key K_e and a ciphertext C outputs $\mathcal{D}_{K_e}\big(C_{[1\ldots|C|-1]}\big)$. It is easy to see that if $\mathcal{SE}$ is IND-CPA secure, then so is $\mathcal{SE}''$ (cf. [12], Proof of IND-CPA security of $\mathcal{SE}_2$, at the end of Section 3). Let $\mathcal{MAC} = (\mathcal{K}_t, \mathcal{T})$ be any UF-CMA secure MAC.

We present an adversary I attacking INT-CTXT security of the scheme described by Construction 3.1 when it uses $\mathcal{SE}''$ and $\mathcal{MAC}$ as the encryption and checksum component schemes. Note that we did not make any assumption about the encoding scheme, so this attack works for any arbitrary encoding scheme. I selects an arbitrary short message M in the message space of the scheme. It queries this message to the encryption oracle and gets back ciphertext C. I then flips the first bit of C and queries the resulting ciphertext $C' = 1\|C_{[1\ldots|C|-1]}$ to the verification oracle.

It is clear that $C' \neq C$, and C' is a valid ciphertext, because $\mathcal{D}''$ ignores the first bit of ciphertext; therefore, $\mathcal{D}''_{K_e}(C') = \mathcal{D}''_{K_e}(C) = M$. Thus, the int-ctxt advantage of I is 1. I makes only one oracle query of length $|M|$, and performs one bit-complementation. □

4.2 Proof of Theorem 3.5

INT-CTXT SECURITY. We will reduce the integrity of ciphertexts of the Modified General Profile to the unforgeability of the underlying MAC scheme. The attack in Proof 1 of Section 4.1 shows that a collision-resistant hash function is not sufficient for integrity of ciphertexts. Moreover, from the attack in Proof 2 of Section 4.1, we know that even an unforgeable MAC cannot provide integrity of ciphertexts if used with any general IND-CPA secure encryption scheme. At a high level, we need an unforgeable MAC and the encryption scheme is required to have the following property for integrity of ciphertexts: for any pair of ciphertexts c, c', if $c \neq c'$ then $m \neq m'$, where m, m' are the corresponding plaintexts. It is easy to see that while CBC with zero IV mode of encryption (or, any other standard deterministic encryption mode) satisfies this property, it may not necessarily hold for any general IND-CPA secure encryption scheme.

We now justify Equation 1. Let I be an adversary attacking the INT-CTXT security of $\mathcal{SE}'$. We construct a forger F breaking the UF-CMA security of $\mathcal{MAC}$. F first runs $\mathcal{K}_e$ to obtain a key K_e for $\mathcal{E}$. It runs I and replies to its queries as follows.

For every encryption oracle query M that I makes, F does the following: It computes $(M_e, M_t) \xleftarrow{\$} \mathit{Encode}(M)$, and then it queries M_t to its own tagging oracle. Let us call the oracle's reply σ. Next, F parses M_e as $M_{el}\|M_{er}$, and forms $M_{el}\|\sigma\|M_{er}$. Then, it computes $C \leftarrow \mathcal{E}_{K_e}(M_{el}\|\sigma\|M_{er})$ and returns C to I.

For every verification oracle query C that I makes, F does the following: It computes $M_{el}\|\sigma\|M_{er} \leftarrow \mathcal{D}_{K_e}(C)$ and $M_t \leftarrow \mathit{Decode}(M_e)$, where $M_e = M_{el}\|M_{er}$. Next, F queries (M_t, σ) to its own verification oracle, then returns 1 to I if the same was returned by its own oracle.

We now analyze F. We claim that F is successful whenever I is successful. First of all, it is straightforward to see that F correctly simulates the encryption oracle for I. Now, if I is successful, then one of its verification oracle queries C' is such that it was not returned by the encryption oracle (i.e. it's new), and its decryption does not return $\perp$. This means that M'_e must be new, where $M'_e (= M'_{el}\|\sigma'\|M'_{er}) \leftarrow \mathcal{D}_{K_e}(C')$, because the base encryption scheme $\mathcal{SE}$ is deterministic (CBC with zero IV). If $(M'_{el}\|\sigma'\|M'_{er})$ is new, then either $M'_{el}\|M'_{er}$ or σ' must be new, which is equivalent to saying that either $M'_{el}\|0^n\|M'_{er} (= M'_t)$ or σ' must be new. This gives rise to two cases. The first case is when M'_t is new (σ' may or may not be new in this case). It is clear that in this case (M'_t, σ') is a valid new message-tag pair. Hence F's verification oracle will return 1.

The second case is when only σ' is new, and M'_t is old, i.e. M'_t is one of the messages that was queried to the tagging oracle. But then in this case, σ' is an invalid tag, as the tagging algorithm is deterministic, and the distinct valid tag was returned as the answer to the corresponding query to the tagging oracle. Hence decryption of C' will return $\perp$.

Hence, the uf-cma advantage of F is the same as the int-ctxt advantage of I. The time complexity of F is basically that of I. F makes the same number of oracle queries as that of I. The total length of all the queries made by F exceeds that of I by only a fixed number of bits, which is the number of queries times $(2n + l - 1)$, due to the use of encoding (at most $n - 1$ bits for padding, n bits for confounder, and l bits for tag).

We now claim the IND-CPA security of $\mathcal{SE}'$. IND-CCA security will then follow from the

IND-CPA security and INT-CTXT security of the scheme.

IND-CPA SECURITY. We show that the composed encryption scheme $\mathcal{SE}'$ is IND-CPA secure if the underlying blockcipher is a PRF and the underlying MAC is a PRF.

Lemma 4.1. For any adversary S attacking IND-CPA security of $\mathcal{SE}'$, that runs in time t, and makes q queries to the left-right encryption oracle, totaling μ bits, there exist adversaries B and G attacking PRF security of E and $\mathcal{MAC}$, such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(S) \leq \mathbf{Adv}^{\text{prf}}_{E}(B) + 2 \cdot \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G) + \frac{\mu^2}{n^2 \cdot 2^n} \ .$$

Furthermore, B runs in time t and makes at most $\lfloor \mu + q \cdot (2n + l - 1)/n) \rfloor$ oracle queries, totaling at most $\mu + q \cdot (2n + l - 1)$ bits; G runs in time t and makes q oracle queries, totaling at most $\mu + q \cdot (2n + l - 1)$ bits.

PROOF OF LEMMA 4.1. We will first show that if the underlying MAC is PRF, then the encryption in the Modified General Profile is similar in terms of security to CBC encryption with random IV (Claim 4.2). Next, from the well known result of [14] (Claim 4.3), we know that CBC encryption with random IV is IND-CPA secure if the underlying block-cipher is a PRF. So, Lemma 4.1 will follow immediately from Claim 4.2 and Claim 4.3. □

Claim 4.2. For any adversary S attacking IND-CPA security of $\mathcal{SE}'$, that runs in time t, and makes q queries to the left-right encryption oracle, totaling μ bits, there exists an adversary D attacking IND-CPA security of CBC encryption scheme with random IV CBC\$ $= (\mathcal{K}_e, \mathcal{E}^{\$}, \mathcal{D}^{\$})$, and an adversary G attacking PRF security of $\mathcal{MAC}$, such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(S) \leq \mathbf{Adv}^{\text{ind-cpa}}_{CBC\$}(D) + 2 \cdot \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G) \ .$$

Furthermore, D runs in time t and makes q queries to the left-right encryption oracle, totaling at most $(\mu + q \cdot (2n + l - 1))$ bits; G runs in time t and makes q oracle queries, totaling at most $(\mu + q \cdot (2n + l - 1))$ bits.

We recall a fact from [14].

Claim 4.3. [**[14], Theorem 4.19**] Let $E\colon\ \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a blockcipher, and let CBC\$ $= (\mathcal{K}_e, \mathcal{E}^{\$}, \mathcal{D}^{\$})$ be the associated CBC encryption scheme with random IV (cf. [9]). Then for any adversary D attacking IND-CPA security of CBC\$, that runs in time t and makes q queries to the left-right encryption oracle, totaling μ n-bit blocks, there exists an adversary B attacking PRF security of E, such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{CBC\$}(D) \leq \mathbf{Adv}^{\text{prf}}_{E}(B) + \frac{\mu^2}{2^n} \ .$$

Furthermore, B runs in time[9] t and makes μ oracle queries, totaling μn bits.

PROOF OF CLAIM 4.2. At a high level the proof follows from the observation that the encoding scheme of Modified General Profile prepends a random confounder to the plaintext. So encrypting this encoded message using CBC with zero IV is equivalent to encrypting any message using CBC with "pseudorandom" IV, because the underlying blockcipher is assumed to be a PRF.

Let S be an adversary attacking IND-CPA security of $\mathcal{SE}'$. For $x \in \{0,1,2,3,4,5\}$, we define the following experiments associated with S.

[9]Due to the difference in convention, this time complexity is different from the one given in [14].

Experiment $\mathbf{ExpH}_x$

$K_e \xleftarrow{\$} \mathcal{K}_e,\ K_t \xleftarrow{\$} \mathcal{K}_t$

Run S replying to its oracle query (M, N) as follows:

$(M_e, M_t) \xleftarrow{\$} \mathit{Encode}(M)$; $(N_e, N_t) \xleftarrow{\$} \mathit{Encode}(N)$; $r \xleftarrow{\$} \{0,1\}^n$

Parse M_e and N_e as $M_{el}\|M_{er}$ and $N_{el}\|N_{er}$, where $|M_{el}| = |N_{el}| = n$

Switch (x):

Case $x = 0$: $\sigma \leftarrow \mathcal{T}_{K_t}(M_t)$; $C \leftarrow \mathcal{E}_{K_e}(M_{el}\|\sigma\|M_{er})$

Case $x = 1$: $C \leftarrow \mathcal{E}_{K_e}(M_{el}\|r\|M_{er})$

Case $x = 2$: $IV\|C \xleftarrow{\$} \mathcal{E}^{\$}_{K_e}(M_{el}\|r\|M_{er})$

Case $x = 3$: $IV\|C \xleftarrow{\$} \mathcal{E}^{\$}_{K_e}(N_{el}\|r\|N_{er})$

Case $x = 4$: $C \leftarrow \mathcal{E}_{K_e}(N_{el}\|r\|N_{er})$

Case $x = 5$: $\sigma \leftarrow \mathcal{T}_{K_t}(N_t)$; $C \leftarrow \mathcal{E}_{K_e}(N_{el}\|\sigma\|N_{er})$

Return C to S.

When S halts and outputs a bit, return that bit.

For $x \in \{0,1,2,3,4,5\}$, let $P_x = \Pr[\,\mathbf{ExpH}_x = 1\,]$ denote the probability that $\mathbf{ExpH}_x$ returns 1. By the definition of $\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(S)$, we have

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(S) = P_5 - P_0 = (P_5 - P_4) + (P_4 - P_3) + (P_3 - P_2) + (P_2 - P_1) + (P_1 - P_0)\ . \qquad (5)$$

We first show that for S, $\mathbf{ExpH}_1$ is indistinguishable from $\mathbf{ExpH}_2$. In $\mathbf{ExpH}_1$, $(M_{el}\|r\|M_{er})$ is encrypted using the CBC mode with zero IV, and the whole ciphertext is returned to S, while in $\mathbf{ExpH}_2$, $(M_{el}\|r\|M_{er})$ is encrypted using the CBC mode with random IV, and the whole ciphertext, except the IV, is returned to S. Note that in the latter case, the ciphertext given to S has the form of $((M_{el} \oplus IV)\|r\|M_{er})$ encrypted using the CBC mode with zero IV. However, since M_{el} and IV are uniformly random strings, to an adversary that doesn't know these in advance, M_{el} and $(M_{el} \oplus IV)$ are indistinguishable. Hence, adversary S cannot distinguish between $\mathbf{ExpH}_1$ and $\mathbf{ExpH}_2$. The same argument applies to show that experiments $\mathbf{ExpH}_3$ and $\mathbf{ExpH}_4$ are indistinguishable in S's view. Hence, $(P_4 - P_3) = (P_2 - P_1) = 0$. Thus, we have

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(S) = (P_5 - P_4) + (P_3 - P_2) + (P_1 - P_0)\ . \qquad (6)$$

Given S, there exist adversaries D and G, such that the following claims hold, and these adversaries use the resources specified in Claim 4.2.

Claim 4.4. $P_3 - P_2 \le \mathbf{Adv}^{\text{ind-cpa}}_{CBC\$}(D)$.

Claim 4.5. $(P_5 - P_4) + (P_1 - P_0) \le 2 \cdot \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G)$.

Equation 6 and the above claims imply Claim 4.2. □

Proof of Claim 4.4. We construct an adversary D breaking the IND-CPA security of CBC\$ using adversary S as follows. For every message-pair query (M, N) that S makes, D first computes $(M_e, M_t) \xleftarrow{\$} \mathit{Encode}(M)$, $(N_e, N_t) \xleftarrow{\$} \mathit{Encode}(N)$, $r \xleftarrow{\$} \{0,1\}^l$. Next, it parses M_e and N_e as $M_{el}\|M_{er}$ and $N_{el}\|N_{er}$, and it queries $(M_{el}\|r\|M_{er}, N_{el}\|r\|N_{er})$ to its own oracle to get back $IV\|C$, where IV is the first ciphertext block. D forwards C back to S. When S halts and returns a bit, D halts and outputs that bit.

We analyze D. The view of S in $\mathbf{ExpH}_2$ is indistinguishable from that in $\mathbf{Exp}^{\text{ind-cpa-0}}_{CBC\$,D}$, and the view of S in $\mathbf{ExpH}_3$ is indistinguishable from that in $\mathbf{Exp}^{\text{ind-cpa-1}}_{CBC\$,D}$. Thus, $P_3 - P_2 \leq \mathbf{Adv}^{\text{ind-cpa}}_{CBC\$}(D)$.

The time complexity of D is basically that of S. D makes the same number of oracle queries as S. The total length of all the queries made by D exceeds that of S by only a fixed number of bits, which is the number of queries times $(2n + l - 1)$, due to the use of encoding (at most $(n-1)$ bits for padding, n bits for confounder, and l bits for tag). □

Proof of Claim 4.5. We construct adversaries G_1 and G_2 breaking the PRF security of $\mathcal{MAC}$ using adversary S such that

$$(P_5 - P_4) + (P_1 - P_0) \leq \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G_2) + \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G_1) \ .$$

G_1 runs $\mathcal{K}_e$ to obtain a key K_e. For every message-pair query (M, N) that S makes, G_1 first computes $(M_e, M_t) \xleftarrow{\$} \mathit{Encode}(M)$. Then it queries M_t to its oracle. Let's call the oracle's reply σ. Next, it parses M_e as $M_{el} \| M_{er}$, forms $M_{el} \| \sigma \| M_{er}$, and computes $C \leftarrow \mathcal{E}_{K_e}(M_{el} \| \sigma \| M_{er})$. G_1 forwards C back to S. When S halts and returns a bit, G_1 halts and outputs the complement bit.

We analyze G_1. When G_1 is in the first experiment of Definition 2.8, then $\sigma = \mathcal{T}_{K_t}(M_t)$, so G_1 simulates $\mathbf{ExpH}_0$ perfectly, and when G_1 is in the second experiment of Definition 2.8, then σ is a random n-bit string, so G_1 simulates experiment $\mathbf{ExpH}_1$ perfectly. Hence, $(P_1 - P_0) \leq \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G_1)$.

Adversary G_2 can be constructed in a similar way, where for every message-pair query (M, N), it does similar things as G_1, but for message N. Thus, we have $(P_5 - P_4) \leq \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G_2)$.

The time complexities of G_1, G_2 are basically that of S. G_1, G_2 make the same number of oracle queries as that of S. The total length of all the queries made by G_1, G_2 exceed that of S by only a fixed number of bits, which is number of queries times $(2n + l - 1)$, due to the use of encoding (at most $(n-1)$ bits for padding, n bits for confounder, and l bits for tag).

Putting G to be one of the adversaries G_1, G_2 with the larger prf-advantage we get

$$\mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G_2) + \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G_1) \leq 2 \cdot \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G) \ .$$

Thus, Claim 4.5 follows. □

IND-CCA security. Equation 1, Lemma 4.1, Theorem 2.4, and Theorem 2.9 imply Equation 2.

4.3 Proof of Theorem 3.8

INT-CTXT security. We will reduce the integrity of ciphertexts of Simplified Profile to the unforgeability of the underlying MAC scheme. First, we note that an attack similar to that in Proof 2 of Section 4.1 can be mounted on Simplified Profile, too. Hence, as pointed out in Section 4.2, for the integrity of ciphertexts it is necessary that the encryption scheme satisfies the following property: for any pair of ciphertexts c, c', if $c \neq c'$ then $m \neq m'$, where m, m' are the corresponding plaintexts. In addition, we point out again that while CBC with zero IV mode of encryption (or, any other standard deterministic encryption mode) satisfies this property, it may not necessarily hold for any general IND-CPA secure encryption scheme.

We justify Equation 3. Let I be an adversary attacking INT-CTXT security of $\mathcal{SE}'$. We construct a forger F breaking the UF-CMA security of $\mathcal{MAC}$. F first runs $\mathcal{K}_e$ to obtain a key K_e for $\mathcal{E}$. It runs I and replies to its queries as follows.

For every encryption oracle query M that I makes, F does the following: It computes $(M_e, M_t) \xleftarrow{\$} \mathit{Encode}(M)$ and then queries M_t to its own tagging oracle. Let us call the oracle's reply σ. Next, F computes $C \leftarrow \mathcal{E}_{K_e}(M_e)$ and returns $C\|\sigma$ to I.

For every verification oracle query $C\|\sigma$ that I makes, F does the following: It computes $M_e \leftarrow \mathcal{D}_{K_e}(C)$ and $M_t \leftarrow \mathit{Decode}(M_e)$. Next, F queries (M_t, σ) to its own verification oracle and returns 1 to I, if the same was returned by its own oracle.

We now analyze F. We claim that F is successful whenever I is successful. First of all, it is straightforward to see that F correctly simulates the encryption oracle for I. Now, if I is successful, then one of its verification oracle queries $C'\|\sigma'$ is such that it was not returned by the encryption oracle (i.e. it's new), and its decryption does not return $\perp$. This gives rise to two cases. The first case is when C' is new (σ' may or may not be new in this case). In this case, $M'_e \leftarrow \mathcal{D}_{K_e}(C')$ must be new, because C' is new, and $\mathcal{SE}$ is deterministic. M'_t is new, because it is equal to M'_e. Thus, (M'_t, σ') is a valid new message-tag pair. Hence F's verification oracle will return 1.

The second case is when only σ' is new and C' is old. However, we show that in this case σ' is invalid, and therefore decryption of C' will return $\perp$. For the same reasons as explained above, old C' implies that M'_t is old, i.e. M'_t is one of the messages which was queried to the tagging oracle. But then σ' is an invalid tag, as the corresponding valid and distinct tag was returned as the answer to the corresponding query.

Hence, the uf-cma advantage of F is the same as the int-ctxt advantage of I. The time complexity of F is basically that of I. F makes the same number of oracle queries as that of I. The total length of all the queries made by F exceeds that of I by only a fixed number of bits, which is the number of queries times $(2n-1)$, due to the use of encoding (at most $(n-1)$ bits for padding, and n bits for confounder).

Before we analyze the IND-CCA security of $\mathcal{SE}'$, let us claim its IND-CPA security.

IND-CPA SECURITY. Theorem 7.1 from [11] states that an encryption scheme composed via the Encode-then-Encrypt&MAC paradigm is IND-CPA if the base encoding scheme is Coll-CPA, the base MAC scheme is PRF, and the base encryption scheme is IND-CPA. However, we cannot use it directly, because the base encryption scheme in Construction 3.7 is CBC with fixed IV, which is obviously not IND-CPA. We present a modification of Theorem 7.1 from [11] to claim the IND-CPA security of the encryption scheme in Construction 3.7, but before that we present the following construction and its security analysis:

Construction 4.6. Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be the CBC encryption scheme with $IV = 0^n$, and $\mathcal{EC} = (\mathit{Encode}, \mathit{Decode})$ be the encoding scheme of Construction 3.7. Then, $\mathcal{SE}'' = (\mathcal{K}_e, \mathcal{E}'', \mathcal{D}'')$ is defined as follows.

- $\mathcal{E}''$ on inputs K_e and M first gets the encodings via $(M_e, M_t) \xleftarrow{\$} \mathit{Encode}(M)$. It then computes $C \leftarrow \mathcal{E}_{K_e}(M_e)$, parses M_e as $\mathit{conf}\|M$, where $|\mathit{conf}| = n$, and returns $\mathit{conf}\|C$.
- $\mathcal{D}''$ on inputs K_e and $\mathit{conf}\|C$ computes $M_e \leftarrow \mathcal{D}_{K_e}(C)$, decodes $(M, M_t) \leftarrow \mathit{Decode}(M_e)$, and returns M.

Claim 4.7. The scheme $\mathcal{SE}''$ defined in Construction 4.6 is as secure as the CBC encryption scheme with random IV, CBC\$ $= (\mathcal{K}_e, \mathcal{E}^\$, \mathcal{D}^\$)$. More precisely, for any adversary A attacking IND-CPA security of $\mathcal{SE}''$, that runs in time t, and makes q queries to the left-right encryption oracle, totaling μ bits, there exists an adversary D attacking IND-CPA security of CBC\$, such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}''}(A) \leq \mathbf{Adv}^{\text{ind-cpa}}_{CBC\$}(D) \ .$$

Furthermore, D runs in time t and makes q queries to the left-right encryption oracle, totaling at most $(\mu + q \cdot (2n-1))$ bits.

PROOF OF CLAIM 4.7. The proof follows from a similar observation (as in the proof of Claim 4.2) that the random confounder prepended to the message by the encoding scheme acts as a "pseudo-random" IV in the encryption, because the underlying blockcipher is assumed to be a PRF.

We construct an adversary D, breaking the IND-CPA security of CBC\$, using adversary A.

For every message-pair query (M, N) that A makes, D first computes $(M_e, M_t) \xleftarrow{\$} \mathit{Encode}(M)$, parses M_e as $\mathit{conf}\|M$, where $|\mathit{conf}| = n$, pads N to multiple block lengths, and computes $N_e \leftarrow \mathit{conf}\|N$. Then it queries (M_e, N_e) to its own oracle and gets back $IV\|C$, where IV is the first ciphertext block. D forwards $(\mathit{conf} \oplus IV)\|C$ back to A. When A halts and returns a bit, D halts and outputs that bit.

We analyze D. We claim that if D is in $\mathbf{Exp}^{\text{ind-cpa-b}}_{CBC\$}(D)$ for $b \in \{0,1\}$, then A's view in the simulated experiment is the same as that in the actual experiment $\mathbf{Exp}^{\text{ind-cpa-b}}_{\mathcal{SE}''}(A)$. Hence $\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}''}(A) \leq \mathbf{Adv}^{\text{ind-cpa}}_{CBC\$}(D)$.

The time complexity of D is basically that of A. D makes the same number of oracle queries as that of A. The total length of all the queries made by D exceeds that of A by only a fixed number of bits, which is the number of queries times $(2n-1)$, due to the use of encoding (at most $(n-1)$ bits for padding, and n bits for confounder) □

From Claim 4.7 and Claim 4.3, we conclude the following.

Claim 4.8. The scheme $\mathcal{SE}''$ defined in Construction 4.6 is IND-CPA secure if the underlying blockcipher E is a PRF. More precisely, for any adversary A attacking IND-CPA security of $\mathcal{SE}''$, that runs in time t and makes q queries to the left-right encryption oracle, totaling μ bits, there exists an adversary B attacking PRF security of E, such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}''}(A) \leq \mathbf{Adv}^{\text{prf}}_{E}(B) + \frac{\mu^2}{n^2 \cdot 2^n} \ .$$

Furthermore, B runs in time t and makes at most $\lfloor (\mu + q \cdot (2n-1)/n) \rfloor$ oracle queries, totaling at most $\mu + q \cdot (2n-1)$ bits.

The following theorem (which is a modification of Theorem 7.1 from [11]) states that the encryption scheme of Construction 3.7 is IND-CPA, if the underlying encoding scheme is Coll-CPA, the underlying MAC scheme is PRF, and the encryption scheme of Construction 4.6 is IND-CPA.

Theorem 4.9. Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$, $\mathcal{MAC} = (\mathcal{K}_t, \mathcal{T})$, and $\mathcal{EC} = (\mathit{Encode}, \mathit{Decode})$ be an encryption scheme, a MAC, and an encoding scheme, respectively, such that the outputs of the encoding scheme are compatible with the inputs to $\mathcal{E}, \mathcal{T}$. Let $\mathcal{SE}'$ and $\mathcal{SE}''$ be the associated encryption schemes as per Construction 3.7 and Construction 4.6, respectively. For any adversary S attacking IND-CPA security of $\mathcal{SE}'$, that runs in time t and makes q queries to the left-right encryption oracle, totaling μ bits, there exist adversaries A attacking IND-CPA security of $\mathcal{SE}''$, G attacking PRF security of $\mathcal{MAC}$, and C attacking Coll-CPA security of $\mathcal{EC}$, such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(S) \leq \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}''}(A) + 2 \cdot \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G) + 2 \cdot \mathbf{Adv}^{\text{coll-cpa}}_{\mathcal{EC}}(C) \tag{7}$$

Furthermore, A and C use the same resources as S, while G runs in time t and makes q oracle queries, totaling at most $(\mu + q \cdot (2n-1))$ bits.

PROOF. The proof is very similar to the Proof of Theorem 7.1 of [11]. Let S be an adversary attacking IND-CPA security of $\mathcal{SE}'$. For $x \in \{1,2,3\}$, we define the following experiments associated with S:

Experiment $\mathbf{ExpH}_x$
  $K_e \xleftarrow{\$} \mathcal{K}_e,\ K_t \xleftarrow{\$} \mathcal{K}_t$
  Run S, replying to its oracle query (M_0, M_1) as follows:
    $(M_{e,0}, M_{t,0}) \xleftarrow{\$} \mathit{Encode}(M_0),\ (M_{e,1}, M_{t,1}) \xleftarrow{\$} \mathit{Encode}(M_1)$
    Switch(x)
      Case $x = 1$: $C \leftarrow \mathcal{E}_{K_e}(M_{e,1}),\ \sigma \leftarrow \mathcal{T}_{K_t}(M_{t,1})$
      Case $x = 2$: $C \leftarrow \mathcal{E}_{K_e}(M_{e,0}),\ \sigma \leftarrow \mathcal{T}_{K_t}(M_{t,1})$
      Case $x = 3$: $C \leftarrow \mathcal{E}_{K_e}(M_{e,0}),\ \sigma \leftarrow \mathcal{T}_{K_t}(M_{t,0})$
    Return $C \| \sigma$ to S.
  Until S halts and returns a bit b.
  Return b.

For $x \in \{1,2,3\}$, let $P_x = \Pr[\,\mathbf{ExpH}_x = 1\,]$ denote the probability that experiment $\mathbf{ExpH}_x$ returns 1. By the definition of $\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(S)$, we have

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(S) = P_1 - P_3 = (P_1 - P_2) + (P_2 - P_3) \tag{8}$$

Given S, there exist adversaries A, G and C, such that the following lemmas hold, and these adversaries use the resources specified in Theorem 4.9.

Lemma 4.10. $P_1 - P_2 \leq \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}''}(A)$.

Lemma 4.11. $P_2 - P_3 \leq 2 \cdot \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G) + 2 \cdot \mathbf{Adv}^{\text{coll-cpa}}_{\mathcal{EC}}(C)$.

Equation 8, and the above lemmas imply Theorem 4.9. □

PROOF OF LEMMA 4.10. We construct an adversary A attacking IND-CPA security of $\mathcal{SE}''$, using the adversary S. A first runs $\mathcal{K}_t$ to obtain a key K_t. For every message-pair query (M_0, M_1) that S makes, A uses that message-pair to query to its own oracle and gets back $\mathit{conf} \| C$. Now, it pads M_1 to multiple block length and computes $M_{t,1} \leftarrow \mathit{conf} \| M_1$, $\sigma \leftarrow \mathcal{T}_{K_t}(M_{t,1})$. It then gives $C \| \sigma$ to S. When S halts and returns a bit b', A halts and returns b'.

If $b = 1$, the adversary A simulates S in the exact same environment as that of $\mathbf{ExpH}_1$. Similarly, if $b = 0$, the adversary A simulates S in the exact same environment as that of $\mathbf{ExpH}_2$. Thus,

$$\begin{aligned} P_1 - P_2 &= \Pr\left[\,\mathbf{Exp}^{\text{ind-cpa-1}}_{\mathcal{SE}''}(A) = 1\,\right] - \Pr\left[\,\mathbf{Exp}^{\text{ind-cpa-0}}_{\mathcal{SE}''}(A) = 1\,\right] \\ &= \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}''}(A). \end{aligned}$$

Adversary A uses the same resources as S. □

PROOF OF LEMMA 4.11. The proof follows directly from Lemma 7.7 and Theorem 7.4 of [11]. □

Below we claim that the encoding scheme $\mathcal{EC}$ in the Simplified profile is Coll-CPA.

Claim 4.12. For any adversary C making q queries to the encoding oracle $\mathcal{EC}(\cdot)$,

$$\mathbf{Adv}^{\text{coll-cpa}}_{\mathcal{EC}}(C) \leq \frac{q(q-1)}{2^{n+1}} \ .$$

PROOF. To justify the claim, we note that *Encode* algorithm prepends a random n-bit confounder to the message, and the only chance that the adversary can make any two encodings M_t, M'_t collide is if any two of q confounders happen to be the same. This can happen with probability at most $\frac{q(q-1)}{2^{n+1}}$, by the well-known birthday bound. □

Theorem 4.9, Claim 4.8, and Claim 4.12 imply the following.

Claim 4.13. The authenticated encryption scheme $\mathcal{SE}'$ described by the Simplified profile (Construction 3.7) is IND-CPA secure, if the underlying blockcipher E is a PRF, and the underlying MAC is a PRF.
Concretely, for any adversary S attacking IND-CPA security of $\mathcal{SE}'$, that runs in time t, and makes q queries to the left-right encryption oracle, totaling μ bits, there exist adversaries B and G attacking PRF security of E and $\mathcal{MAC}$, respectively, such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(S) \leq \mathbf{Adv}^{\text{prf}}_{E}(B) + 2 \cdot \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(G) + \frac{q(q-1)}{2^{n+1}} + \frac{\mu^2}{n^2 \cdot 2^{n+1}} \ .$$

Furthermore, B runs in time t and makes at most $\lfloor (\mu + q \cdot (2n-1)/n) \rfloor$ oracle queries, totaling at most $\mu + q \cdot (2n-1)$ bits; G runs in time t and makes q oracle queries, totaling at most $\mu + q \cdot (2n-1)$ bits.

IND-CCA SECURITY. Equation 3, Claim 4.13, Theorem 2.4, and Theorem 2.9 imply Equation 4.

5 Conclusions

We took a close look at the two designs of authenticated encryption in Kerberos version 5, called General and Simplified Profiles. We show that the authenticated encryption paradigm used in General profile does not provide integrity, even if it uses secure building blocks(e.g. a secure hash function and a secure encryption scheme). While our attacks do not apply for particular instantiations of the General Profile suggested in the specifications, they do show limitation of the design. We suggest simple and easy to implement modifications, and we show that the resulting scheme provably provides privacy and authenticity, under standard assumptions. We prove that Modified General Profile and Simplified Profile are IND-CCA and INT-CTXT secure, if they utilize secure building blocks. This justifies the assumption about the security of encryption necessary for the recent formal-methods-based symbolic analyses. Together, these results provide strong security guarantees for Kerberos that we believe will help its standardization, and will emphasize importance of formal security analysis of practical protocols.

6 Acknowledgments

We thank Ken Raeburn and Sam Hartman for clarifications on Kerberos specifications, Bogdan Warinschi for useful discussions, the anonymous reviewers of 2007 IEEE Symposium on Security

and Privacy for their helpful comments, Anupam Datta and Kenneth Paterson for comments on the preliminary draft and Brandon Craig for editing English. Alexandra Boldyreva is supported in part by NSF CAREER award 0545659. Virendra Kumar is supported in part by the mentioned grant of the first author.

References

[1] M. Backes, I. Cervesato, A. D. Jaggard, A. Scedrov, and J.-K. Tsay. Cryptographically Sound Security Proofs for Basic and Public-Key Kerberos. In *ESORICS '06*. Springer, 2006.

[2] M. Backes and B. Pfitzmann. Symmetric Encryption in a Simulatable Dolev-Yao Style Cryptographic Library. In *CSFW '04*. IEEE, 2004.

[3] M. Backes, B. Pfitzmann, and M. Waidner. A composable cryptographic library with nested operations. In *CCS '03*. ACM, 2003.

[4] M. Backes, B. Pfitzmann, and M. Waidner. Symmetric Authentication within a Simulatable Cryptographic Library. In *ESORICS '03*. Springer, 2003.

[5] G. Bella and L. C. Paulson. Kerberos Version 4: Inductive Analysis of the Secrecy Goals. In *ESORICS '98*. Springer, 1998.

[6] G. Bella and E. Riccobene. Formal Analysis of the Kerberos Authentication System. *Journal of Universal Computer Science*, 3(12):1337–1381, 1997.

[7] M. Bellare. New Proofs for NMAC and HMAC: Security Without Collision-Resistance. CRYPTO '06, 2006.

[8] M. Bellare, R. Canetti, and H. Krawczyk. Keying Hash Functions for Message Authentication. In *CRYPTO '96*. Springer, 1996.

[9] M. Bellare, A. Desai, E. Jokipii, and P. Rogaway. A Concrete Security Treatment of Symmetric Encryption. In *FOCS '97*, pages 394–403. IEEE, 1997.

[10] M. Bellare, J. Kilian, and P. Rogaway. The security of the cipher block chaining message authentication code. In *CRYPTO '04*. Springer, 2004.

[11] M. Bellare, T. Kohno, and C. Namprempre. Breaking and provably repairing the SSH authenticated encryption scheme: A case study of the Encode-then-Encrypt-and-MAC paradigm. *ACM Trans. Inf. Syst. Secur.*, 7(2):206–241, 2004.

[12] M. Bellare and C. Namprempre. Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. *J. Cryptology*, 21(4):469–491, 2008.

[13] M. Bellare and P. Rogaway. Encode-then-encipher encryption: How to exploit nonces or redundancy in plaintexts for efficient cryptography. In *ASIACRYPT '00*. Springer, 2000.

[14] M. Bellare and P. Rogaway. An Introduction to Modern Cryptography. UCSD CSE 207 Course Notes, 2005. Available at `http://www.cse.ucsd.edu/~mihir/cse207/index.html`.

[15] M. Bellare and P. Rogaway. The Game-Playing Technique and its Application to Triple Encryption. In *EUROCRYPT*, 2006.

[16] S. M. Bellovin and M. Merritt. Limitations of the Kerberos authentication system. *SIGCOMM Comput. Commun. Rev.*, 20(5):119–132, 1990.

[17] F. Butler, I. Cervesato, A. D. Jaggard, and A. Scedrov. A Formal Analysis of Some Properties of Kerberos 5 Using MSR. In *CSFW '02*. IEEE, 2002.

[18] F. Butler, I. Cervesato, A. D. Jaggard, A. Scedrov, and C. Walstad. Formal Analysis of Kerberos 5. In *Theoretical Computer Science*, 2006.

[19] D. Dolev and A. C. Yao. On the security of public key protocols. IEEE Transactions on Information Theory, IT-29(12), 1983.

[20] J. T. Kohl. The use of encryption in Kerberos for network authentication (invited). In *CRYPTO '89*. Springer, 1989.

[21] T. Kohno. Authenticated Encryption in Practice: Generalized Composition Methods and the Secure Shell, CWC, and WinZip Schemes. UCSD Dissertation, 2006.

[22] H. Krawczyk. The Order of Encryption and Authentication for Protecting Communications (or: How Secure Is SSL?). In *CRYPTO '01*, pages 310–331. Springer, 2001.

[23] D. Micciancio and B. Warinschi. Completeness Theorems for the Abadi-Rogaway Logic of Encrypted Expressions. *Journal of Computer Security*, 12(1):99–129, 2004.

[24] C. Neuman, T. Yu, S. Hartman, and K. Raeburn. The Kerberos Network Authentication Service (V5). *Network Working Group. Request for Comments: 4120*, 2005. Available at `http://www.ietf.org/rfc/rfc4120.txt`.

[25] K. Raeburn. Advanced Encryption Standard (AES) Encryption for Kerberos 5. *Network Working Group. Request for Comments: 3962*, 2005. Available at `http://www.ietf.org/rfc/rfc3962.txt`.

[26] K. Raeburn. Encryption and Checksum Specifications for Kerberos 5. *Network Working Group. Request for Comments: 3961*, 2005. Available at `http://www.ietf.org/rfc/rfc3961.txt`.

[27] S. G. Stubblebine and V. D. Gligor. On Message Integrity in Cryptographic Protocols. In *Symposium on Security and Privacy '92*. IEEE, 1992.

[28] X. Wang, D. Feng, X. Lai, and H. Yu. Collisions for Hash Functions MD4, MD5, HAVAL-128 and RIPEMD. ePrint Archive: Report 2004/199, 2004. Available at `http://eprint.iacr.org/`.

[29] T. D. Wu. A Real-World Analysis of Kerberos Password Security. In *NDSS '99*. The Internet Society, 1999.

[30] T. Yu. The Kerberos Network Authentication Service (Version 5). IETF Internet draft. Request for Comments: 1510, 2006.

[31] T. Yu, S. Hartman, and K. Raeburn. The Perils of Unauthenticated Encryption: Kerberos Version 4. In *NDSS '04*. The Internet Society, 2004.