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
{aboldyre,virendra}@cc.gatech.edu

Abstract

Kerberos is a widely-deployed network authentication protocol that is being considered for standardization. Many works have analyzed its security, identifying flaws and often suggesting fixes, thus helping the protocol's evolution. Several recent results present successful formal-methods-based verification of a significant portion of the current version 5, and some even imply security in the computational setting. For these results to be meaningful, encryption in Kerberos should satisfy strong cryptographic security notions. However, neither currently deployed as part of Kerberos encryption schemes nor their proposed revisions are known to provably satisfy such notions. We take a close look at Kerberos' encryption and confirm that most of the options in the current version provably provide privacy and authenticity, some with slight modification that we suggest. Our results complement the formal-methods-based analysis of Kerberos that justifies its current design.

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 big universities and corporations, is part of all major computing platforms such as Windows (starting from Windows 2000), Linux and Mac OS, and is a draft standard at IETF [28].

Security of Kerberos has been analyzed in many works, e.g. [16, 27, 6, 5, 25, 20, 29]. 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 [22, 24, 23].

*A preliminary shortened version of this paper appears in IEEE Security & Privacy 2007 Proceedings.

What is certainly desirable for the upcoming standard is to provide some guarantees that the protocol does not only resist some *specific* known attacks, but withstands a very large class of possible attacks, under some accepted 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 the 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 parts. However, a known limitation of such analyses is 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, which is 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 the cryptographic primitives used in the protocol need to satisfy strong notions of security (in the computational setting). Namely, the encryption scheme utilized by the protocol needs to provide privacy against chosen-ciphertext attacks (be IND-CCA secure) and also authenticity and integrity of ciphertexts (be INT-CTXT secure) [2, 1].

However, it is not known whether encryption[1] in Kerberos is IND-CCA and INT-CTXT. Certain known vulnerabilities indicate that encryption in version 4 did not satisfy these notions [29]. While encryption in the current version 5 is designed to resist known attacks it is not clear whether it *provably* resists all attacks of the class, and if yes – under which assumptions. Provable security has become a de-facto standard approach in modern cryptographic research. Cryptographers design plenty of cryptographic schemes for almost all imaginable future applications, and usually provide rigorous proofs of security for their constructions. It is somewhat surprising that the schemes that are actually used in deployed protocols remain unanalyzed from the provable-security perspective. Our work aims at closing this gap.

1.2 Contributions

We take a close look at the encryption schemes used in Kerberos v.5 (according to its specifications [24, 23]) in order to prove them secure, in the IND-CCA and INT-CTXT sense, assuming the underlying building blocks (e.g. a block cipher) are secure.

GENERAL PROFILE. We first look at the encryption scheme description in the current version 5 specification (cf Section 6 in [24].) We will refer to it as "General profile". Fix a block cipher with input-output length n and a key for it. A message M is first padded so that its length is multiple of n. Next a random n-bit string $conf$ is chosen. Then a checksum, i.e. a hash function is applied to the string $conf \parallel 0^n \parallel M$. Let us call the checksum's output σ. Finally, the block cipher in the CBC mode with fixed initial vector $IV = 0^n$ is applied to the string $conf \parallel \sigma \parallel M$. Decryption is defined accordingly. The recommended options are DES as the block cipher and MD4 or MD5 as hash functions, which is not a very good choice for known reasons. DES is an outdated standard since its key and block sizes are too small given modern computing power, and collisions have been found in MD4 and MD5 [26]. But what we show is that even if one assumes the "right" component options such as a secure block cipher in a secure encryption mode and a secure hash function, the construction is not secure in general. That is, there exist attacks on the scheme composed of certain secure components that show that it does not provide integrity of

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

ciphertexts. We note that these attacks do not apply if the particular encryption scheme recommended in [24] is used. Nevertheless the attacks show a weakness in the design.

MODIFIED GENERAL PROFILE. We propose simple, easy to implement modifications that are sufficient for security of the design of the General profile. Namely, we show that if a message authentication code (MAC) that is secure, i.e. unforgeable against chosen-message attacks (UF-CMA), is used as a checksum in place of the hash function, and it is applied to $0^{2n} \parallel M$ instead of $conf \parallel 0^n \parallel M$, then the Modified general profile encryption scheme is IND-CCA and INT-CTXT, also assuming that the underlying block cipher is secure, i.e. a pseudorandom function (PRF). In particular, AES and HMAC [8] are good candidates for a block cipher and MAC respectively.

SIMPLIFIED PROFILE. Next we look at recently-proposed revisions to the encryption design in Kerberos aka. Simplified profile (cf. Section 5 in [24] and [23]). This encryption scheme, that did not catch up with implementations yet, recommends to use AES or Triple-DES as a block cipher 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 is pre-pended. The block cipher in CBC mode or a variant of CBC mode with ciphertext-stealing 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. This confirms soundness of the design of the Simplified profile, that, unlike General profile, is secure in general.

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 code (MAC) schemes. They show that the only one out of three most straight-forward 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 paradigm 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. But for some reasons Kerberos uses some variations of Encrypt-and-MAC or MAC-then-Encrypt methods that also rely on the properties of the encodings of the message, i.e. of pre-processing of the message before encryption and MAC are applied.

Bellare et al. [11] analyze security of encryption in another widely deployed protocol, Secure Shell aka. SSH. They suggest several modifications to the SSH encryption to fix certain flaws and 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 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 [24]) 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 IND-CPA secure base encryption scheme but as we mentioned above the base encryption in Kerberos is CBC with fixed IV and is not IND-CPA secure.

1.4 Outline

After defining some notation we recall the relevant cryptographic primitives and their security definitions. Next we outline the designs of authentication schemes in the General and Simplified profile authenticated encryption schemes of Kerberos' specification, and the modification to the General profile we propose. We follow with detailed security analysis of the schemes and 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 $X \,\|\, Y$ 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 multiple of n bits for some integer n, $X[i]$ denotes its ith block, meaning $X = X[1]||\ldots||X[l]$ where $l = |X|/n$ and $|X[i]| = n$ for all $i = 1, \ldots, l$. If S is a set then $X \xleftarrow{\$} S$ 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, possibly on some inputs. 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 input the secret key K and a plaintext $M \in \mathrm{MsgSp}$ 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 a 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 $M \in \mathrm{MsgSp}$.

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 knows 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 $\mathrm{atk} \in \{\mathrm{cpa}, \mathrm{cca}\}$, adversary A and a bit b define the experiments $\mathbf{Exp}^{\text{ind-atk-b}}_{\mathcal{SE},A}$ as follows. In all experiments first the secret key K is generated by $\mathcal{K}$. Let LR be the oracle that on input M_0, M_1, b returns M_b. The adversary A is given access to *left-right encryption oracle* $\mathcal{E}_K(\mathrm{LR}(\cdot,\cdot,b))$ that A can query on any pair of messages in MsgSp and of of equal length. 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, and the experiment returns d as well. The *ind-atk-advantage* of a adversary A is defined as:

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

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[2].

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. 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 (or be INT-CTXT secure) if any adversary 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 random key K is generated by $\mathcal{K}$. The adversary has access to the encryption oracle $\mathcal{E}_K(\cdot)$. It is successful if it can output a valid ciphertext C (i.e. $\mathcal{D}_K(C) \neq \perp$) that was never returned by the encryption oracle.

It has been shown [12] that if an encryption scheme is IND-CPA and INT-CTXT then it is also IND-CCA. To achieve INT-CTXT security encryption scheme often utilize message authentication codes (MACs), that we define below.

MESSAGE AUTHENTICATION CODE (MAC).

Definition 2.4 [MAC] A *message authentication code (MAC)* $\mathcal{MAC} = (\mathcal{K}, \mathcal{T})$ with associated *message space* MsgSp is defined by two algorithms:

- The randomized *key generation* algorithm $\mathcal{K}$ returns a a secret key K.
- The deterministic[3] *mac aka. tagging* algorithm $\mathcal{T}$ takes input the secret key K and a plaintext M to return a mac aka. 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 no adversary can forge a valid tag for a new message.

[2] 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 makes 2^{60} basic operations in some fixed model of computation), and does 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}.)

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

Definition 2.5 [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 the tagging oracle $\mathcal{T}_K(\cdot)$. It is successful if it can output a message-tag pair (M, σ) such that $M \in \text{MsgSp}$, σ is a valid tag for M under K, and M was not queried to the tagging oracle.

Another (stronger) security definition requires that the output of the MAC is indistinguishable from a random string.

Definition 2.6 [PRF] 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 $\mathcal{T}$. The MAC is called *pseudorandom function 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

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

It is known that any MAC that is PRF is also UF-CMA.

HASH FUNCTION.

Definition 2.7 [Hash function] A hash function $HF = (\mathcal{K}, \mathcal{H})$ consists of two algorithms. The *key generation* algorithm $\mathcal{K}$ outputs a key K[4]. The *hash* algorithm $\mathcal{H}$ on inputs K and $M \in \{0,1\}$ outputs the hash value H.

Definition 2.8 [Collision-resistance] A hash function $HF = (\mathcal{K}, \mathcal{H})$ is called *collision-resistant* if every adversary with reasonable resources who is given a random K output by $\mathcal{K}$ can output two messages $M_1, M_2 \in \{0,1\}$ such that $\mathcal{H}_K(M_1) = \mathcal{H}_K(M_2)$, $M_1 \neq M_2$ only with 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 or random nonce, etc.

Definition 2.9 [Encoding scheme] An *encoding scheme* $\mathcal{EC} = (Enc, Dec)$ with associated *message space* MsgSp is defined by two algorithms. The (possibly) randomized or stateful *encoding* algorithm Enc takes a message $M \in \text{MsgSp}$ 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, let $(M_e, M_t) \xleftarrow{\$} Enc(M)$ and $(M', M'_t) \leftarrow Dec(M_e)$ then, the consistency condition requires that $M = M'$ and $M_t = M'_t$.

The following is from [13, 11].

Definition 2.10 [Coll-CPA] Let $\mathcal{EC} = (Enc, Dec)$ be an encoding scheme. It is called *collision-resistant against chosen-plaintext attacks or Coll-CPA* if every adversary A with reasonable resources has only small success probability , called the *coll-cpa-advantage* of A, $\mathbf{Adv}^{\text{coll-cpa}}_{\mathcal{EC}}(A)$ in the following experiment. The adversary has access to the encoding oracle $Enc(\cdot)$ and it is considered successful if it ever gets two replies containing M_t and M'_t such that $M_t = M'_t$.

[4] Our results can also be applied to keyless hash functions.

PSUEDORANDOM FUNCTION FAMILY. A family of functions is a map $E\colon \{0,1\}^k \times \{0,1\}^m \to \{0,1\}^n$, where we regard $\{0,1\}^k$ as the *keyspace* for the function family in that a *key* $K \in \{0,1\}^k$ induces a particular function from this family, which we denote by $E_K(\cdot)$.

Definition 2.11 [PRF] Let $E\colon \{0,1\}^k \times \{0,1\}^m \to \{0,1\}^n$ be a function family. Let R be the set of all functions from $\{0,1\}^m$ to $\{0,1\}^n$. E is called *pseudorandom function 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

$$\mathbf{Adv}^{\mathrm{prf}}_E(A) \;=\; \Pr\left[\, K \xleftarrow{\$} \{0,1\}^k \;:\; A^{E_K(\cdot)} = 1 \,\right] - \Pr\left[\, g \xleftarrow{\$} R \;:\; A^{g(\cdot)} = 1 \,\right] .$$

3 Analysis of Encryption in Kerberos v.5

3.1 General Profile

We first look at the encryption scheme specified in [24]. 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-MAC-then-Encrypt] Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$, $\mathcal{EC} = (Enc, Dec)$, $\mathcal{MAC} = (\mathcal{K}_m, \mathcal{T})$ be an encryption scheme, an encoding scheme and a checksum (i.e. MAC or hash function). The message space of the corresponding *Encode-then-MAC-then-Encrypt* scheme $\mathcal{SE}' = (\mathcal{K}', \mathcal{E}', \mathcal{D}')$ is that of is $\mathcal{EC}$ and the rest of the algorithms are defined as follows.

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

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

The next construction specifies in more detail how Kerberos' encryption operates. Figure 1 illustrates the design.

Construction 3.2 [Authenicated encryption in Kerberos. General profile] Let $E\colon \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher. Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be the CBC encryption mode (cf. e.g. [9] for the formal description) with IV fixed to be a string of n zeros[5]. Let $\mathcal{MAC} = (\mathcal{K}_m, \mathcal{T})$ describe a hash function with output of length l bits, which is keyless or whose key is public. Let $\mathcal{EC} = (Enc, Dec)$ be an encoding scheme such that Enc with $\mathrm{MsgSp} = \{0,1\}^*$ on input M pads it to make the length of $l + |M|$ multiple of n bits (so that decoding is unambiguous), picks a random confounder of n bits $conf \xleftarrow{\$} \{0,1\}^n$,

[5]The Kerberos' specification also allows the stateful update of the IV, i.e. the IV is assigned to be the last block of the previous ciphertext. Our analyses applies to this case as well. But since this option is not commonly used, we do not consider it in detail. We note however, that [24] 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.

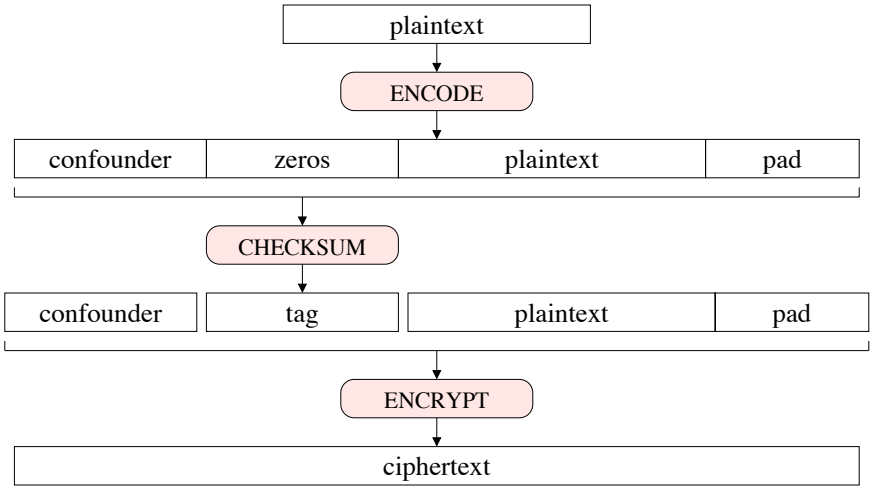

Figure 1: Encryption in Kerberos v.5. General profile.

computes $M_e \leftarrow conf \parallel M$ and $M_t \leftarrow conf \parallel 0^n \parallel M$ and returns (M_e, M_t). Dec 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[6].

SECURITY ANALYSIS OF GENERAL PROFILE. As we noted in the Introduction the recommended instantiations with DES and MD4 or MD5 are not good choices. DES is an outdated standard since its key and block sizes are too small given modern computing power, and collisions have been found in MD4 and MD5 [26]. But what our results show, is that using the "right" building blocks such as for example, AES and a collision-resistant hash function will not necessarily solve the problem. The reason is that the Encode-then-MAC-then-Encrypt composition method does not provide integrity in general, when it uses a hash function as checksum, even if it uses a secure encryption option for the underlying encryption scheme.

Theorem 3.3 Let $\mathcal{EC} = (Enc, Dec)$ be the encoding scheme from 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-MAC-then-Encrypt (Construction 3.1) does not provide integrity (is not INT-CTXT secure). Concretely, there exists an adversary I with reasonable resources with $\mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}'}(I)$ being 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 (with the corresponding secret key being secret, of course), but in this case the attack makes use of a rather artificial IND-CPA scheme. We note that the attacks we provide in the proof does not translate into an attack on any of the recommended options. It just shows a limitation in the general design.

MODIFIED GENARAL PROFILE. We now suggest simple, easy-to-implement modifications to the General Profile construction, and show that they are sufficient to prove security of the scheme. Namely we

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

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 the above paragraph, because now we rely on a particular encryption scheme the General profile uses. We now define the construction and state its security.

Construction 3.4 [Modified General profile] The construction is like Construction 3.2, except that $\mathcal{MAC} = (\mathcal{K}_m, \mathcal{T})$ is a message authentication code and the encoding algorithm Enc of $\mathcal{EC}$ computes M_t as $0^{n+l} \parallel M$ (as opposed to $conf \parallel 0^l \parallel M$ for a random string $conf$, as before).

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 block cipher is a PRF and the $\mathcal{MAC}$ is UF-CMA secure.
Concretely, let $\mathcal{SE}$, $\mathcal{EC}$ and $\mathcal{MAC}$ be an encryption scheme, an encoding scheme and a checksum respectively. Let $\mathcal{SE}'$ be the authenticated encryption scheme associated to them by Modified General profile (Construction 3.4). Then for any adversary I attacking INT-CTXT security of $\mathcal{SE}'$ that runs in time at most t and asks at most q queries, totalling at most μ n-bit blocks there exist adversaries F, B attacking UF-CMA security of $\mathcal{MAC}$ and PRF security of E respectively, such that

$$\mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}'}(I) \leq \mathbf{Adv}^{\text{uf-cma}}_{\mathcal{MAC}}(F) + \mathbf{Adv}^{\text{prf}}_{E}(B) + 2^{-n}. \tag{1}$$

Furthermore, F runs in time at most $t_F \approx t + O(q)$ and asks at most $q_F = q$ queries; B runs in time at most $t_B \approx t + O(q)$ and asks at most $q_B = 3q + \mu$ queries.
And for any adversary A attacking IND-CCA security of $\mathcal{SE}'$ that runs in time at most t and asks at most q queries, totalling at most μ n-bit blocks there exist adversaries F, B attacking UF-CMA security of $\mathcal{MAC}$ and PRF security of E respectively, such that

$$\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}'}(A) \leq 2 \cdot \mathbf{Adv}^{\text{uf-cma}}_{\mathcal{MAC}}(F) + 3 \cdot \mathbf{Adv}^{\text{prf}}_{E}(B) + \frac{2 + \mu^2}{2^n}. \tag{2}$$

Furthermore, F runs in time at most $t_F \approx t + O(q)$ and asks at most $q_F = q$ queries; B runs in time at most $t_B \approx t + O(q + n(\mu + O(q(n + l))))$ and asks at most $q_B = q + \mu$ queries.

The proof is in Section 4.2 .

AES that is believed to be a PRF and HMAC [8] that is proven to be UF-CMA secure if its underlying hash function is collision resistant, constitute good instantiations for the above design.

3.2 Simplified Profile

Designers of Kerberos had the right intuition that the General profile's design is not particularly strong and proposed a new design that they call "Simplified profile" (cf. Section 5 in [24] and [23]). Again we start with a more general composition method that outlines the design.

Construction 3.6 [Encode-then-Encrypt&MAC] Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$, $\mathcal{MAC} = (\mathcal{K}_m, \mathcal{T})$, $\mathcal{EC} = (Enc, Dec)$ be an encryption scheme, a MAC and an encoding scheme such that the outputs of the encoding scheme are compatible with message spaces to $\mathcal{E}, \mathcal{T}$. 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 $\mathcal{K}_e, \mathcal{K}_m$ and returns their outputs $K_e \parallel K_m$.
- $\mathcal{E}'$ on inputs $K_e \parallel K_m$ and M first gets the encodings via $(M_e, M_t) \xleftarrow{\$} Enc(M)$. It then computes $C \xleftarrow{\$} \mathcal{E}_{K_e}(M_e)$, $\sigma \leftarrow \mathcal{T}_{K_m}(M_t)$ and returns $C \parallel \sigma$.
- $\mathcal{D}'$ on inputs $K_e \parallel K_m$ and $C \parallel \sigma$ computes $M_e \leftarrow \mathcal{D}_{K_e}(C)$, decodes $(M, M_t) \leftarrow Dec(M_e)$, computes $\sigma' \leftarrow \mathcal{T}_{K_m}(M_t)$ and returns M if $\sigma = \sigma'$, and $\perp$ otherwise.

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

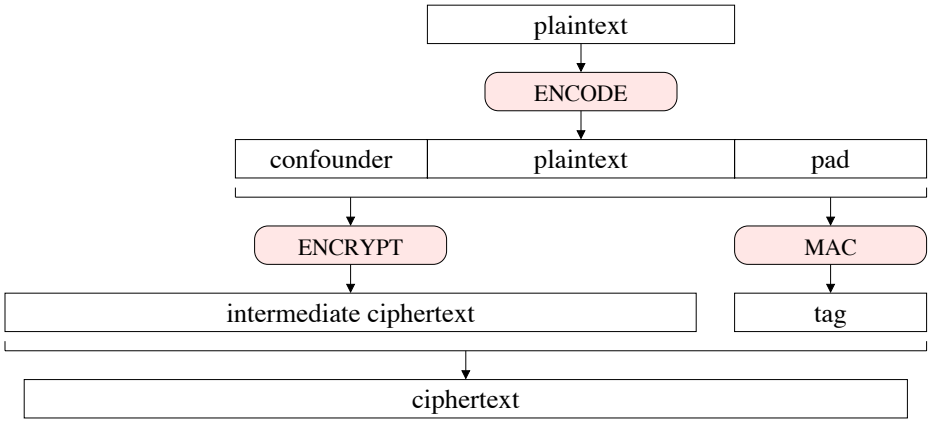

Figure 2: Authenticated encryption in Kerberos v.5. Simplified profile.

Construction 3.7 [Authenicated encryption in Kerberos. Simplified profile] Let $E\colon \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher. Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be the CBC encryption mode with $IV = 0^n$. Let $\mathcal{MAC} = (\mathcal{K}_m, \mathcal{T})$ be a MAC, let $\mathcal{EC} = (Enc, Dec)$ be an encoding scheme such that Enc on input M pads M to make its length multiple of n bits (while permitting unambiguous decoding), picks a random confounder of n bits $conf \xleftarrow{\$} \{0,1\}^n$, computes $M_e \leftarrow conf \parallel M$ and $M_t \leftarrow conf \parallel M$ and returns (M_e, M_t). Dec 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 block cipher E is a PRF and the message authentication code $\mathcal{MAC}$ is a PRF.
Concretely, let $\mathcal{SE}$, $\mathcal{EC}$ and $\mathcal{MAC}$ be an encryption scheme, an encoding scheme and a checksum respectively. 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 at most t and asks at most q queries, totalling at most μ n-bit blocks 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 at most $t_F \approx t + O(q)$ and asks at most $q_F = q$ queries.
And for any adversary A attacking IND-CCA security of $\mathcal{SE}'$ that runs in time at most t and asks at most q queries, totalling at most μ n-bit blocks there exist adversaries D, B attacking PRF security of $\mathcal{MAC}$ and PRF security of E respectively, such that

$$\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}'}(A) \leq 4 \cdot \mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(D) + \mathbf{Adv}^{\text{prf}}_{E}(B) + \frac{2q + \mu^2}{2^n}. \quad (4)$$

Furthermore, D runs in time at most $t_D \approx t + O(q)$ and asks at most $q_D = q$ queries; B runs in time at most $t_B \approx t + O(q + n\sigma)$ and asks at most $q_B = \mu$ queries.

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 (cf. Section 6 [14]), so PRF security is a sufficient assumption.

AES is believed to be a PRF, Triple DES was shown to be a PRF in the ideal cipher model [15] and HMAC was proven to be a PRF [7] assuming the underlying compression hash function is a PRF (cf. [7] for the definition of the latter), therefore they are the right choices of instantiations for the Simplified profile.

4 Proofs

4.1 Proof of Theorem 3.3

We present two alternative proofs of insecurity of the General profile authenticated encryption. The first proof considers the General profile that uses a natural encryption scheme and not-so-natural hash function as checksum. The second proof considers the General profile that uses a special encryption scheme, but the checksum can be instantiated with arbitrary secure MAC.

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

Let $HF = (\mathcal{K}, \mathcal{H})$ be a hash function which hashes messages in $bits$ to l bits and is believed to be collision resistant. Consider a modified hash function $HF' = (\mathcal{K}, \mathcal{H}')$, where $\mathcal{H}'$ on input K and M outputs $M_0 \parallel \mathcal{H}(K, M_{1...|M|-1})_{1...l-1}$, where M_0 is the first bit of M, $M_{1...|M|-1}$ is M minus first bit of M and $\mathcal{H}(K, M)_{1...l-1}$ is $\mathcal{H}(K, M)$ minus the first bit of $\mathcal{H}(K, M)$. We show that HF' is also a collision resistant hash function.

Assume we have an adversary A who can find collisions in HF'. We then construct an adversary B who finds collisions in HF. B gives its own challenge key K to A and gets back two messages M, N. B computes $M' \leftarrow 0\|M$, $N' \leftarrow 0\|N$ and outputs M', N'.

If $\mathcal{H}(K,M) = \mathcal{H}(K,N)$ and $M \neq N$, then tt is easy to see that $\mathcal{H}'(K,M') = \mathcal{H}'(K,N')$. This is because $\mathcal{H}'(K,M') = 0\|H(K,M)_{1\ldots l-1}$ and $\mathcal{H}'(K,N') = 0\|H(K,N)_{1\ldots l-1}$.

B is almost as efficient as A. Hence if HF is collision resistant then so is HF'.

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

We claim that int-ctxt advantage of A is 1. This is justified as follows. Consider $conf\|\sigma\|M = \mathcal{D}_K(ctr\|C)$. Here $\sigma = \mathcal{H}'(K,M_t)$ and $M_t = conf\|0^l\|M$. So, $\sigma = (conf_0\|\mathcal{H}(K,M_{t1\ldots|M_t|-1})_{1\ldots l-1})$. $ctr\|C$ can be parsed as $ctr\|C[1]\|C[2]\|D$ where $C[1]$ and $C[2]$ are first and second blocks of C, and D is the remaining part of C. From the CTR description $C[1] = conf \oplus E_K(\langle ctr+1\rangle)$ and $C[2] = (conf_0\|\mathcal{H}(K,M_{t1\ldots|M_t|-1})_{1\ldots n-1}) \oplus E_K(\langle ctr+2\rangle)$. 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 $C'[1] = (\overline{conf_0}\|conf_{1\ldots n-1}) \oplus E_K(\langle ctr+1\rangle)$ and $C'[2] = (\overline{conf_0}\|\mathcal{H}(K,M_{t1\ldots|M_t|-1})_{1\ldots n-1}) \oplus E_K(\langle ctr+2\rangle)$ Since $C' = C'[1]\|C'[2]\|D$, $\mathcal{D}_{K_e}(C') = (M'_{el}\|\sigma'\|M'_{er})$, and $M'_{el} = (\overline{conf_0}\|conf_{1\ldots n-1})$, $M'_{er} = M$ and $\sigma' = (\overline{conf_0}\|\mathcal{H}(K,M_{t1\ldots|M_t|-1})_{1\ldots l-1})$. Now notice that $M'_e = (M'_{el}\|M'_{er}) = (\overline{conf_0}\|conf_{(1\ldots n-1)})\|M$ and $M'_t = (M'_{el}\|0^l\|M'_{er}) = (\overline{conf_0}\|conf_{1\ldots n-1})\|0^l\|M$. It is clear $M_{t1\ldots|M_t|-1} = M'_{t1\ldots|M'_t|-1}$ since, M_t and M'_t differ only in first bit.

So, $\sigma' = (\overline{conf_0}\|\mathcal{H}(K,M_{t1\ldots|M_t|-1})_{1\ldots l-1}) = (\overline{conf_0}\|\mathcal{H}(K,M'_{t1\ldots|M'_t|-1})_{1\ldots l-1}) = \mathcal{H}'(K,M'_t)$. Thus, (M'_t,σ') is a valid message-tag pair. Hence, $N\|C'$ is a valid ciphertext that was never returned by the encryption oracle and so the int-ctxt advantage of A is 1. A makes one oracle query of length n bits and performs two operations of bit-complementation.

PROOF 2. 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 K and M outputs $0\|E_K(M)$ and $\mathcal{D}'$ on input C outputs $D_K(C_{1\ldots|C|-1})$. It is easy to see that $\mathcal{SE}'$ is IND-CPA secure if $\mathcal{SE}$ is. Please refer to Proof of Proposition 3.4 of [12] for the detailed proof. Let $\mathcal{MAC} = (\mathcal{K}_m,\mathcal{T})$ be any UF-CMA secure MAC.

We present an adversary A attacking INT-CTXT security of the scheme described by Construction 3.1 when it uses $\mathcal{SE}'$ and $\mathcal{MAC}$ as the encryption and MAC component schemes. A 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. A then flips the first bit of C and returns the resulting ciphertext $C' = 1\|C_{1\ldots|C|-1}$.

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

4.2 Proof of Theorem 3.5

INT-CTXT SECURITY. We justify Equation 1. We construct an adversary (a forger) F breaking the UF-CMA security of $\mathcal{MAC}$. F first runs $\mathcal{K}_e$ to obtain a key K_e. For every query M that I makes, F computes $(M_e,M_t) \xleftarrow{\$} Enc(\cdot)$ and then queries M_t to its own oracle. Let us call the oracle's reply σ. Next F parses M_e as $M_{el}\|M_{er}$ and uses σ returned by its oracle to form $M_{el}\|\sigma\|M_{er}$. Then, it computes $C \leftarrow \mathcal{E}_{K_e}(M_{el}\|\sigma\|M_{er})$ and returns C to I. When I outputs s a new ciphertext C', F computes $M'_{el}\|\sigma'\|M'_{er} \leftarrow \mathcal{D}_{K_e}(C')$ and $M'_t \leftarrow 0^{2n}\|M'_{er}$. Finally F returns (M'_t,σ').

We now analyze F. C' being new and $\mathcal{SE}$ being deterministic ensures that $(M'_{el}\|\sigma'\|M'_{er})$ is new, meaning at least one of M'_{el}, σ' and M'_{er} must be new. Let BadM be an event associated with I in which C' output by I is valid and new, M'_{el} is new but σ' and M'_{er} are not new. We now state the following claims.

Let B be the adversary attacking the PRF security of underlying block cipher E. We state the following lemmas.

Claim 4.1 $\mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}'}(I) \leq \mathbf{Adv}^{\text{uf-cma}}_{\mathcal{MAC}}(F) + \Pr[\,\mathsf{BadM}\,]$

Claim 4.2 For any adversary I attacking the INT-CTXT security of $\mathcal{SE}'$ and creating BadM event that runs in time at most t and asks at most q queries, totalling at most μ n-bit blocks there exists an adversary attacking PRF security of underlying block cipher E such that

$$\Pr[\,\mathsf{BadM}\,] \leq \mathbf{Adv}^{\text{prf}}_{E}(B) + 2^{-n}.$$

Furthermore, B runs in time at most $t_B \approx t + O(q)$ and asks at most $q_B = (3q + \mu)$ queries.

Then Equation 1 follows from Lemma 4.1 and Lemma 4.2 after observing that F makes 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 number of queries times $(n + l)$, due to the use of encoding as explained above. The time complexity of F differs from I by just the amount of time required to execute $Enc(\cdot)$ and $\mathcal{E}_{K_e}(\cdot)$ for every query and $\mathcal{K}_e$ once. In addition, F uses $\mathcal{D}_{K_e}(\cdot)$ and parsing once to compute a valid message-tag pair from the ciphertext returned by I.

Proof of Claim 4.1 Let GoodC be an event associated with I in which C' output by I is valid and new. Then by observing that I int-ctxt advantage is $\Pr[\,\mathsf{GoodC}\,]$ and by conditioning we get

$$\begin{aligned}
\mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}'}(I) &= \Pr[\,\mathsf{GoodC}\,] \\
&= \Pr[\,\mathsf{GoodC} \mid \neg\mathsf{BadM}\,] \cdot \Pr[\,\neg\mathsf{BadM}\,] + \Pr[\,\mathsf{GoodC} \mid \mathsf{BadM}\,] \cdot \Pr[\,\mathsf{BadM}\,] \\
&\leq \Pr[\,\mathsf{GoodC} \mid \neg\mathsf{BadM}\,] + \Pr[\,\mathsf{BadM}\,]
\end{aligned}$$

We now claim that

$$\mathbf{Adv}^{\text{uf-cma}}_{\mathcal{MAC}}(F) \geq \Pr[\,\mathsf{GoodC} \mid \neg\mathsf{BadM}\,]\ .$$

This is because GoodC given $\neg\mathsf{BadM}$ implies that either σ or M'_{er} are new. If σ' is new, then $\mathcal{MAC}$ being deterministic implies M'_t must be new, $M'_t \leftarrow 0^{2n}\|M'_{er}$ so if M'_{er} is new and thus M'_t must be new. If M'_{er} is new then M'_t must be new. This implies that F's output is a successful forgery.

F makes 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 number of queries times $(n + l)$, due to the use of encoding as explained above. The time complexity of F differs from I by just the amount of time required to execute $Enc(\cdot)$ and $\mathcal{E}_{K_e}(\cdot)$ for every query and $\mathcal{K}_e$ once. In addition, F uses $\mathcal{D}_{K_e}(\cdot)$ and parsing once to compute a valid message-tag pair from the ciphertext returned by I. Thus adversary F is almost as efficient as I.

Hence, we get the statement of the claim. ∎

Proof of Claim 4.2 We construct an adversary B breaking PRF security of E. For simplicity we consider only one block message query by I. It is easy to see that the proof can be generalized for arbitrary size message queries. B runs $\mathcal{K}_m$ to obtain a key K_m. For every query m_i that I makes, B first computes $(M_{ie}, M_{it}) \xleftarrow{\$} Enc(m_i)$. Then, it computes $\sigma_i \leftarrow \mathcal{T}_{K_m}(M_{it})$. Next, it parses M_{ie} as $conf_i \| m_i$ and forms a three-block message $conf_i \| \sigma_i \| m_i$. Let us denote $conf_i$, σ_i and m_i by $M_i[1]$, $M_i[2]$ and $M_i[3]$ respectively. $C_i[0] \leftarrow IV \leftarrow 0^n$. For $j = 1, 2, 3$ it queries $(C_i[j-1] \oplus M_i[j])$ to its oracle and gets back $C_i[j]$ one by one. Finally, it gives back $C_i \leftarrow C_i[1] \| C_i[2] \| C_i[3]$ to I. When I outputs a new ciphertext C' then it parses it as $C'[1] \| C'[2] \| C'[3]$ and queries its oracle at $(C'[1] \oplus M_i[2])$ for all $i = 1, 2, ..., q$ and compares the oracle's output with $C'[2]$. If anyone of them matches then B halts and outputs 1 otherwise outputs 0.

We analyze B. We claim that I's view in the simulated experiment is the same as that in the int-ctxt experiment. It is clear that B follows the same steps as in Construction 3.4 to obtain a ciphertext. In particular for any message m_i it follows CBC mode of encryption with zero IV for encrypting $M_i[1] \| M_i[2] \| M_i[3]$ which is exactly what is done in Construction 3.4. Let $M'[1] \| M'[2] \| M'[3]$ be the plaintext corresponding to C' and let $M'[2]$ be same as $M_l[2]$, for any l from 1 to q (occurence of BadM event). Now, from CBC mode of encryption with zero IV it is clear that

$C'[1] = E_K(M'[1])$

$C'[2] = E_K(C'[1] \oplus M'[2])$

But, $M'[2] = M_l[2]$ so $C'[2] = E_K(C'[1] \oplus M_l[2])$

Thus, if the adversary B is in world 1 then the ciphertext output by A can always be used to break the PRF security of E. In world 0 the adversary B will fail with a maximum probability of 2^{-n} when $C'[2]$ matches with a random output by oracle. Hence, $\mathbf{Adv}^{\mathrm{prf}}_E(B) \geq \Pr[\,\mathsf{BadM}\,] - 2^{-n}$.

Basic intuition is that since $M'[1]$ is new so $C'[1]$ is also new. Therefore, $(M_l[2] \oplus C'[1])$ is a new block of message which was never queried to B's oracle but, B can compute $E_K(M_l[2] \oplus C'[1])$ (which is equal to $C'[2]$) from ciphertext returned by I. So, he can break the PRF security of E.

Let q, μ the number of queries and total length of queries (n-bit blocks) made by I then B makes at most $(3q+\mu)$ number of oracle queries. The total length of all the queries made by B exceeds that of I by $3q$ n-bit blocks. The time complexity of B differs from A by just the amount of time required to execute $Enc(\cdot)$, $\mathcal{T}_{K_m}(\cdot)$, parsing and XORing for every query and running $\mathcal{K}_m$ once. Since these operations are efficient, adversary B is almost as efficient as I.

Hence, we get the statement of the claim. ∎

Before proving IND-CCA security, let us asses IND-CPA security.

IND-CPA SECURITY. We show that the composed encryption scheme $\mathcal{SE}'$ is at least as secure as cipher block chaining mode of encryption with random IV aka. CBC$.

Theorem 4.3 [[14], Section 4] $E\colon \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher and let CBC$ be the CBC encryption scheme with random IV (cf. [9]). Then for any adversary D attacking IND-CPA security of CBC$ that runs in time at most t and asks at most q queries, totalling at most μ n-bit blocks there exists an adversary B attacking PRF security of E such that

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

Furthermore, B runs in time at most $t_B \approx t + O(q + n\sigma)$ and asks at most $q_B = \mu$ queries.

Claim 4.4 For any adversary S attacking IND-CPA security of $\mathcal{SE}'$ that runs in time at most t and asks at most q queries totalling μ n-bit blocks there exists an adversary D attacking IND-CPA security of CBC$ such that

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

Furthermore, D runs in time at most $t_D \approx t + O(q)$ and asks at most $q_D = q$ queries.

Proof: D runs $\mathcal{K}_m$ to obtain a key K_m. For every message-pair query (M, N) that S makes, D first computes $(M_e, M_t) \xleftarrow{\$} Enc(M)$ and $(N_e, N_t) \leftarrow Enc(M)$. Then, it computes $\sigma_M \leftarrow \mathcal{T}_{K_m}(M_t)$ and $\sigma_N \leftarrow \mathcal{T}_{K_m}(N_t)$. Next, it parses M_e and N_e as $M_{el}\|M_{er}$ and $N_{el}\|N_{er}$ and queries $((M_{el}\|\sigma_M\|M_{er}), (N_{el}\|\sigma_N\|N_{er}))$ 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. We claim that S's view in the simulated experiment is the same as that in the actual experiment $\mathbf{Exp}^{\text{ind-cpa-b}}_{\mathcal{SE}',S}$. The reason is simple. The first block of an encoded message is a random confounder. So, when an encoded message is encrypted using CBC$ it is similar to encrypting the message using $\mathcal{SE}'$ except that in the latter case the first block of message encrypted using underlying CBC encryption mode with zero IV would be confounder XORed with IV. But, since confounder and IV are random so it is indistinguishable for the adversary S. Hence, if $IV\|C$ is the output of CBC$ applied on an encoded message then C is indistinguishable from the output of $\mathcal{SE}'$ applied on the corresponding message. So, the probability that D outputs 1 in $\mathbf{Exp}^{\text{ind-cpa-b}}_{CBC\$,D}$ is same as the probability that S would output 1 in $\mathbf{Exp}^{\text{ind-cpa-b}}_{\mathcal{SE}',S}$ for any b.

D makes same number of oracle queries as that of S. The total length of all the queries made by D exceeds that of S by only a fixed number of bits which is number of queries times $(n + l)$, due to the use of encoding and tagging as explained above. The time complexity of D differs from S by just the amount of time required to execute $Enc(\cdot)$ and $\mathcal{T}_{K_m}(\cdot)$ and parsing for every query and running $\mathcal{K}_m$ once. ∎

Theorem 4.3 and Claim 4.4 immediately imply the following.

Theorem 4.5 $E\colon \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher and $\mathcal{SE}'$ be the authenticated encryption scheme of Modified General Profile (Construction 3.4). Then any adversary S attacking IND-CPA security of $\mathcal{SE}'$ that runs in time at most t and asks at most q queries, totalling μ n-bit blocks there exists an adversary B attacking PRF security of E such that

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

Furthermore, B runs in time at most $t_B \approx t + O(q + n(\mu + O(q(n + l))))$ and asks at most $q_B = \mu$ queries.

IND-CCA SECURITY.

Theorem 4.6 **[[12], Theorem 3.2]** Let $\mathcal{SE}$ 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 IND-CCA security

of $\mathcal{SE}$ there exist adversaries B, C attacking the scheme's IND-CPA and IND-CCA security respectively such that

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

Furthermore, the resources of B and C are same as those of A.

Equation 1, Theorem 4.5 and Theorem 4.6 imply Equation 2. ∎

4.3 Proof of Theorem 3.8

INT-CTXT SECURITY. We justify Equation 3.

Let I be some 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 for every query M that I makes, F does the following. It computes $(M_e, M_t) \xleftarrow{\$} Enc(M)$. It then queries M_t to its own oracle and gets back σ. It then computes $C \leftarrow \mathcal{E}_{K_e}(M_e)$ and returns $C\|\sigma$ to I. At some point I outputs a ciphertext $C'\|\sigma'$. F computes $M'_e \leftarrow \mathcal{D}_{K_e}(C')$, $M'_t \leftarrow Dec(M'_e)$ and returns (M'_t, σ').

We now analyze F. If I is successful then $C'\|\sigma'$ is new, meaning at least one part must be new. This gives rise to two cases. The first case is when C' is new (σ' may or may not be new in this case). The we claim that F's output is a valid forgery. Note that M'_e as $\mathcal{D}_{K_e}(C')$ must be new because C' is new and $\mathcal{SE}$ is deterministic. M'_t is new because equal to M_e. Thus, (M'_t, σ') is a valid new message-tag pair.

The second case is when only σ' is new and C' is old. But we show that in this case σ' is invalid. For the same reasons as explained above old C' implies that M'_t is old, i.e. 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 same as the int-ctxt advantage of I. Let q be the number of oracle queries I does, and let μ be their total length. Then F also makes q queries. The total length of all the queries made by F is $\mu + qn$. The time complexity of F is almost as that of I.

Before accessing 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 a PRF and the base encryption scheme is IND-CPA:

Theorem 4.7 [[11], Theorem 7.1] Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$, $\mathcal{MAC} = (\mathcal{K}_m, \mathcal{T})$, $\mathcal{EC} = (Enc, Dec)$ be an encryption scheme, a MAC and an encoding scheme such that the outputs of the encoding scheme are compatible with message spaces to $\mathcal{E}, \mathcal{T}$. For any adversary S attacking IND-CPA security of the associated authenticated encryption scheme $\mathcal{SE}'$ build via Construction 3.6, there exist adversary A attacking IND-CPA security of $\mathcal{SE}$, an adversary D attacking PRF security of $\mathcal{MAC}$ and adversary C attacking Coll-CPA security of $\mathcal{EC}$ such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(S) \leq \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A) + 2\mathbf{Adv}^{\text{prf}}_{\mathcal{MAC}}(D) + 2\mathbf{Adv}^{\text{coll-cpa}}_{\mathcal{EC}}(C) \tag{5}$$

and A, D, and C use the same resources as S except that As and Ds inputs to their respective oracles may be of different lengths than those of S (due to the encoding).

Claim 4.8 The encoding scheme $\mathcal{EC}$ in the Simplified profile is Coll-CPA. I.e. for adversary C making q queries to its oracle

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

To justify the claim we note that Enc 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 the confounders happen to be same. This can happen with probability at most $q/2^n$.

We assume that the base MAC scheme is PRF. However, we cannot yet claim IND-CPA security of the Simplified profile, because its base encryption scheme is CBC with fixed IV which is obviously not IND-CPA. We note, however, that Theorem 7.1 in [11] also holds for the case when the encryption scheme, whose encryption algorithm is applied to encoded message is IND-CPA. The details are as follows.

Construction 4.9 Let $E\colon \{0,1\}^k \times \{0,1\}^m \to \{0,1\}^n$ be a block cipher and let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be the associated CBC encryption mode with $IV = 0^n$, $\mathcal{EC} = (Enc, Dec)$ 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 $(M_e, M_t) \xleftarrow{\$} Enc(M)$. It then computes $C \xleftarrow{\$} \mathcal{E}_{K_e}(M_e)$, parses M_e as $conf \| M$ and returns $conf \| C$.
- $\mathcal{D}''$ on inputs K_e and $conf \| C$ computes $M_e \leftarrow \mathcal{D}_{K_e}(C)$, decodes $(M, M_t) \leftarrow Dec(M_e)$ and returns M.

Claim 4.10 The scheme $\mathcal{SE}''$ defined in Construction 4.9 is IND-CPA secure if the underlying block cipher is a PRF. More precisely, for any adversary A attacking IND-CPA security of $\mathcal{SE}''$ that runs in time at most t and asks at most q queries, totalling at most μ n-bit blocks, there exists an adversary B attacking PRF security of E such that

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

Furthermore, B runs in time at most $t_B \approx t + O(q + n\sigma)$ and ask at most $q_B = q$ queries.

The proof follows from the proof of Theorem 4.5 and the CBC$ security statement in Section 4 [14].

We now claim that Lemma 7.6 from [11] holds when the encryption scheme in question is $\mathcal{SE}''$ defined above. We provide the modifications to the construction of the adversary adversary A breaking the IND-CPA security of the underlying encryption scheme $\mathcal{SE}'' = (\mathcal{K}_e, \mathcal{E}'', \mathcal{D}'')$ using adversary S (cf. the proof Lemma 7.6, [11] for details). A first runs $\mathcal{K}_m$ once to obtain a key K_m. For every message-pair query (M, N) that S makes, A uses that message-pair to query to its own oracle and gets back $conf \| C$. Now it pads N to multiple block length and computes $N_t \leftarrow conf \| N$, $\sigma_N \leftarrow \mathcal{T}_{K_m}(N_t)$. It then gives $C \| \sigma_N$ to S. When S halts and returns a bit, A halts and outputs that bit. The rest of the analysis of the proof Lemma 7.6 and Theorem 7.1 in [11] holds.

Equation 5, Claim 4.8 and Claim 6 imply the following.

Claim 4.11 The authenticated encryption scheme $\mathcal{SE}'$ described by the Simplified profile (Construction 3.7) is IND-CPA secure if the underlying block cipher E is a PRF and the message authentication code

$\mathcal{MAC}$ is a PRF.
Concretely, for any adversary S attacking IND-CPA security of $\mathcal{SE}'$ that runs in time at most t and asks at most q queries, totalling at most μ n-bit blocks there exist adversaries B, D attacking PRF security of E and $\mathcal{MAC}$ respectively, such that

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

Furthermore, B runs in time at most $t_B \approx t + O(q + n\mu)$ and asks at most $q_B = \mu$ queries; D runs in time at most $t_D \approx t$ and asks at most $q_D = q$ queries.

IND-CCA SECURITY. Equation 3, Claim 4.11 and Theorem 4.6 imply Equation 4.

5 Conclusions

We took a close look at two designs of authenticated encryption in Kerberos version 5 called General and Simplified profiles. We show that General profile does not provide integrity even if it uses secure building blocks such as a secure hash function and encryption scheme. While our attack does not apply for particular instantiations of the General profile suggested in specifications, it shows a limitation of the design. We suggest simple and easy-to-implement modifications an show that the resulting scheme provably provides privacy and authenticity under standard assumptions. are IND-CCA and INT-CTXT secure if they utilize secure building blocks. This justifies the assumption about 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 for their helpful comments, and Anupam Datta for comments on the preliminary draft. 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*. 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. Authenticated encryption in SSH: provably fixing the SSH binary packet protocol. In *CCS '02*. ACM, 2002.

[12] M. Bellare and C. Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In *ASIACRYPT '00*. Springer, 2000.

[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. `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 Using. In *Theoretical Computer Science*, 2006.

[19] D. Dolev and A. C. Yao. On the security of public key pro- tocols. 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. Searchable symmetric encryption: Improved definitions and efficient constructions. UCSD Dissertation, 2006.

[22] C. Neuman, T. Yu, S. Hartman, and K. Raeburn. The Kerberos network authentication service (v5). *Network Working Group. Request for Comments: 4120. Available at* `http://www.ietf.org/rfc/rfc4120.txt`, 2005.

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

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

[25] S. G. Stubblebine and V. D. Gligor. On message integrity in cryptographic protocols. In *Symposium on Security and Privacy '92*. IEEE, 1992.

[26] 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/`.

[27] T. D. Wu. A real-world analysis of Kerberos password security. In *NDSS '99*. The Internet Society, 1999.

[28] T. Yu. The Kerberos network authentication service (version 5). IETF Internet draft. Request for Comments: 1510, 2006.

[29] T. Yu, S. Hartman, and K. Raeburn. The perils of unauthenticated encryption: Kerberos version 4. In *NDSS '04*. The Internet Society, 2004.