

A Formal Analysis of Some Properties of Kerberos 5 Using MSR

Frederick Butler, Iliano Cervesato,
Aaron D. Jaggard, and Andre Scedrov

Supported by ONR URI

Project Goals

- ◆ Give precise statement and formal analysis of a real world protocol
 - Find a real world protocol - Kerberos 5
 - Use MSR for formalization and analysis
- ◆ Identify and formalize protocol goals
- ◆ Give proofs of achieved protocol goals
 - Gain experience in reasoning with MSR
- ◆ Note any anomalous behavior
 - Consider possible fixes, test these

Background

◆ Kerberos 4

- Analyzed using inductive approach (Bella & Paulson)

◆ Kerberos 5

- Simplified version analyzed with Mur ϕ (Mitchell, Mitchell, & Stern)

◆ MultiSet Rewriting (MSR)

- Partially supported by ONR MURI

Our Previous Achievements

- ◆ Formalizations of large fragments of Kerberos 5
 - Using MSR + extensions
- ◆ Formal analysis of protocol
 - Proofs of protocol properties
 - Curious behavior seen
- ◆ Interactions with Kerberos designers
 - Our work seems helpful, interested in future results

Recent Achievements

- ◆ **Theorems for final protocol exchange**
 - End goal of participants
 - Confidentiality of shared key
 - Authentication of different credentials
- ◆ **Theorems in detailed formalization**
 - Extended proof methods to this setting
 - These add detail to abstract theorems and proofs
- ◆ **A new anomaly**
 - Curious behavior involving ticket options

Introduction

Kerberos Overview

Two Views of Kerberos 5

Protocol Properties

Anomalies

Protocol Goals and History

◆ Protocol goals

- Repeatedly authenticate a client to multiple servers
- Does not guard against DOS attacks

◆ Kerberos 4 - 1989

◆ Kerberos 5

- Specified in RFC 1510 (1993)
- Subsequent revisions

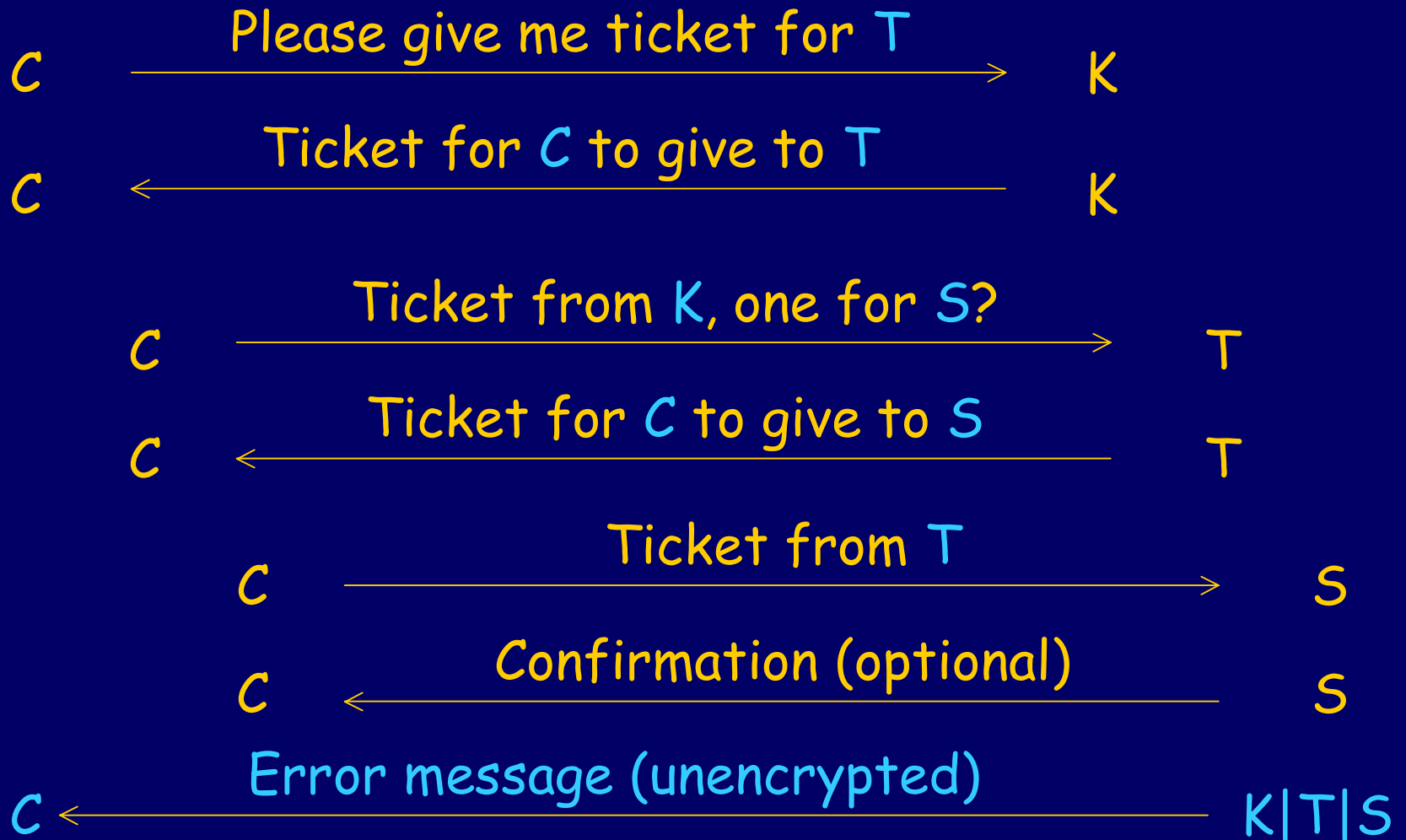
◆ A real world protocol

- Windows 2000 (RFC 1510 + extensions)
- User login, file access, printing, etc.

Kerberos 5

- ◆ Client C wants ticket for end server S
 - Tickets are encrypted - unreadable by C
- ◆ C first obtains long term (e.g., 1 day) ticket from a Kerberos Authentication Server K
 - Makes use of C 's long term key
- ◆ C then obtains short term (e.g., 5 min.) ticket from a Ticket Granting Server T
 - Based on long term ticket from K
 - C sends this ticket to S

Protocol Messages



Introduction

Kerberos Overview

Two Views of Kerberos 5

Protocol Properties

Anomalies

Abstract Formalization

◆ Contains core protocol

- Other formalization refines this one
- Enough detail to prove authentication and confidentiality theorems

◆ Exhibits an anomaly

- This appears to be structural and not due to omitted detail

Detailed Formalization

◆ Uses richer message structure

- Adds some fields for options
- Models encryption type
- Adds checksums

◆ Exhibits anomalies

- Encryption type option specific to this level
- Structural anomaly also seen at abstract level
 - Anonymous ticket option used in variation of this
- Ticket option anomaly specific to this level

MSR Overview

◆ Use Typed MSR + some extensions

- Network messages and principal knowledge/states captured by "facts"
- Facts are predicates in language (with \exists) corresponding to the protocol
- Protocol run is a sequence of multisets of facts
 - Obtain this sequence by successive application of rewriting rules (subject to constraints) which capture actions of principals and Dolev-Yao intruder

Introduction

Kerberos Overview

Two Views of Kerberos 5

Protocol Properties

Anomalies

Properties Proved

	Confidentiality	Authentication
Ticket Granting Exchange	Abstract Abstract Detailed	Abstract Abstract Detailed
Client/Server Exchange	Abstract	Abstract

Proof Methods

- ◆ Inspired by work of Schneider
- ◆ Define functions on MSR facts
 - k-Rank - data origin authentication
 - E-Corank - secrecy
- ◆ Proofs
 - State desired property
 - Find applicable (co)rank functions
 - Determine effect of MSR rules on these functions

Abstract Authentication Theorem

- ◆ If T processes the message

$$\{k_{CT}, C\}_{k_T}, \{C\}_{k_{CT}}, C, S, n_2$$

then some K created k_{CT} and sent

$$C, \{k_{CT}, C\}_{k_T}, \{k_{CT}, n_1, T\}_{k_C}$$

and C sent *some*

$$X, \{C\}_{k_{CT}}, C, S', n'_2$$

- ◆ In Kerberos 4, C *must* have sent the ticket and *not* the generic X (Bella & Paulson)
- ◆ Similar theorem for Client/Server exchange
 - Ticket came from T , authenticator from C

Detailed Authentication Theorem

- ◆ Add details to abstract level theorem to obtain a theorem in our detailed formalization
- ◆ If T processes the message

$\{TFlags, k_{CT}, C\}_{k_T}, \{C, ck, t\}_{k_{CT}}, TOpts, C, S, n_2, e$
then some K created k_{CT} and sent

$C, \{TFlags, k_{CT}, C\}_{k_T}, \{k_{CT}, n_1, TFlags, T\}_{k_C}$
and C sent *some*

$X, \{C, ck, t\}_{k_{CT}}, TOpts', C, S', n'_2, e'$
with

$$ck = [TOpts', C, S', n'_2, e']_{k_{CT}}$$

An Authentication Theorem

◆ Authenticate data origin using rank

- Show ticket $\{TFlags, k_{CT}, C\}_{k_T}$ originates with some K
- Show authenticator $\{C, ck, t\}_{k_{CT}}$ originates with C
 - Relies on the confidentiality of k_{CT} , proved using corank
- Abstract level proofs follow same outline

Introduction

Kerberos Overview

Two Views of Kerberos 5

Protocol Properties

Anomalies

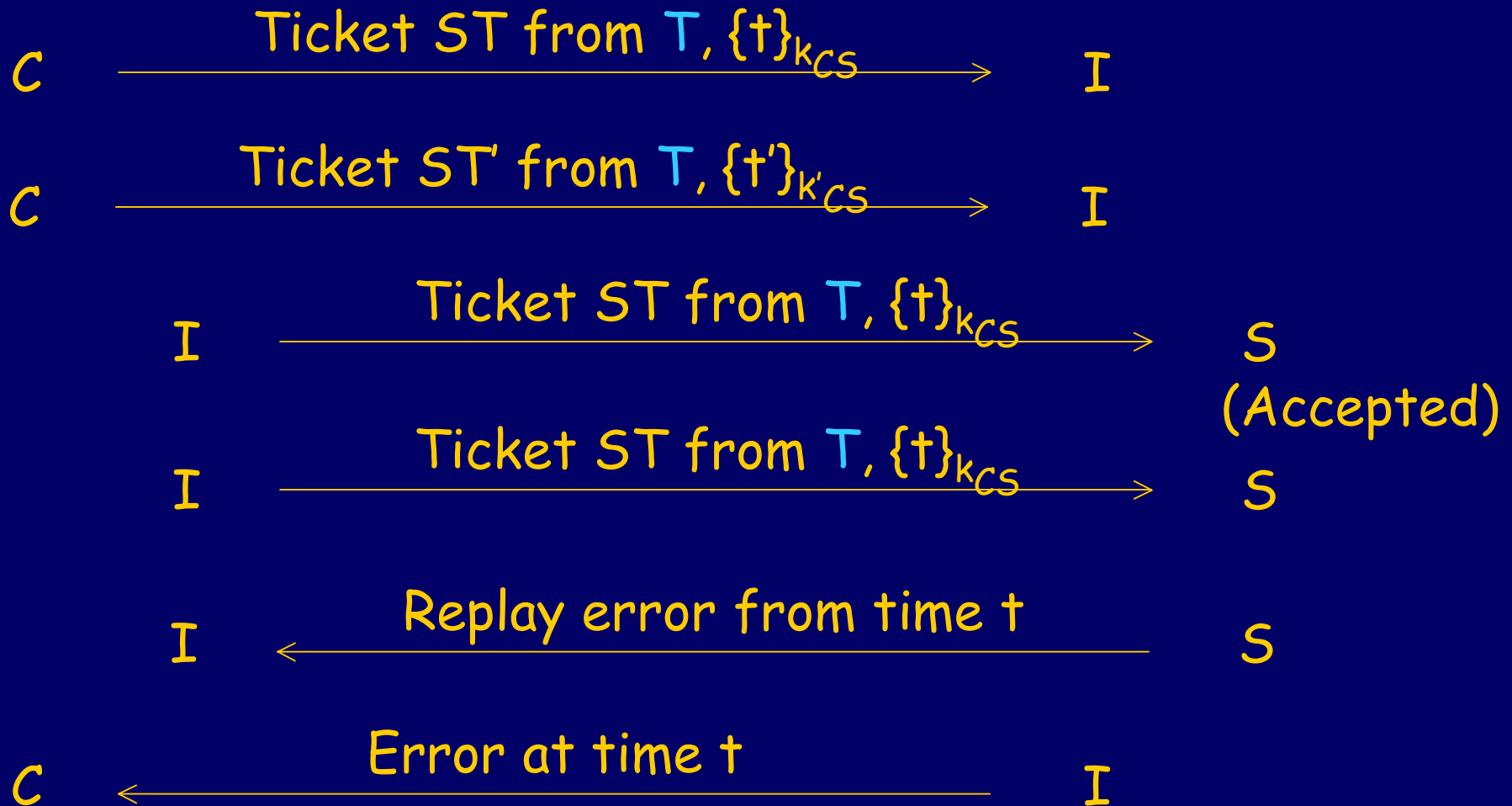
Old Anomalies

- ◆ Interesting curiosities, but don't appear dangerous
- ◆ Encryption type anomaly
 - Difficult to recover from lost long term key
- ◆ Ticket switch anomaly
 - Client has incorrect beliefs about data in her possession
 - Application to anonymous tickets
 - Anonymous option under review

Ticket Option Anomaly

- ◆ C obtains tickets ST and ST' with different options for use with server S
- ◆ C does not request mutual authentication from S , so she does not expect a response if the requests are successfully processed
- ◆ Assume that S can detect replays
 - Saves authenticators in a cache (following RFC 1510)

Ticket Option Anomaly



Ticket Option Anomaly

- ◆ C 's request at time t was accepted, but her request at time t' was never seen by S
- ◆ C sees an error message with the timestamp t
 - Might assume request at t not accepted, request at t' accepted
 - I uses the replay to unpack the encrypted timestamp t
 - S 's use of a replay cache allows this to occur
- ◆ Effects are similar to those of ticket switch found before but for more ticket options
 - Replay cache not yet formalized

Conclusions

- ◆ Formalizations of Kerberos 5 at different levels of detail
 - Extended MSR to do this
 - MSR can handle real world protocols
- ◆ Proofs of properties which hold here
 - Parallel theorems and proofs in two formalizations
 - Authentication and confidentiality throughout
 - Gained additional experience in reasoning with MSR
- ◆ Anomalous behavior
- ◆ Interactions with Kerberos designers

Future Work

- ◆ Continue interaction with Kerberos designers
- ◆ Formalizations
 - Add structure and functionality
 - Public key extensions
- ◆ Analysis
 - Investigate temporal checks
 - Properties in more detailed formalizations
 - Anomalies - what can we still prove? Fix? Accept?
 - Systematic generation of rank functions
- ◆ Explore use of automated tools