Internet Engineering Task Force (IETF)
Request for Comments: 5868
Category: Informational
ISSN: 2070-1721

S. Sakane
K. Kamada
S. Zrelli
Yokogawa Electric Corp.
M. Ishiyama
Toshiba Corp.

May 2010

Problem Statement on the Cross-Realm Operation of Kerberos

Abstract

This document provides background information regarding large-scale Kerberos deployments in the industrial sector, with the aim of identifying issues in the current Kerberos cross-realm authentication model as defined in RFC 4120.

This document describes some examples of actual large-scale industrial systems, and lists requirements and restrictions regarding authentication operations in such environments. It also identifies a number of requirements derived from the industrial automation field. Although they are found in the field of industrial automation, these requirements are general enough and are applicable to the problem of Kerberos cross-realm operations.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Kerberos Version 5 is a widely deployed mechanism that enables a server to authenticate a client before granting it access to a certain service. Each client belongs to a managed domain called a realm. Kerberos supports authentication when a client and a server belong to different realms. This is called cross-realm authentication.

There exist several ways for using Kerberos in large-scale distributed systems. Such infrastructures are typically split into several managed domains for geographical reasons, and to implement different management policies. In order to ensure smooth network operations in such systems, a common authentication mechanism for the different managed domains is required. When using the Kerberos cross-realm operation in large-scale distributed systems, some issues arise.

As industrial automation is moving towards wider adoption of Internet standards, the Kerberos authentication protocol represents one of the best alternatives for ensuring the confidentiality and the integrity of communications in control networks while meeting performance and security requirements. However, the use of Kerberos cross-realm operations in large-scale industrial systems may introduce issues that could cause performance and reliability problems.

This document briefly describes the Kerberos Version 5 system and its cross-realm operation mode. Then it describes two case-study systems that Kerberos could be applied to, and describes seven requirements in those systems (in terms of both management and operations) that outline various constraints that Kerberos operations might be subjected to. Finally, it lists six issues related to Kerberos cross-realm operations when applied to those systems.

Note that this document might not describe all issues related to Kerberos cross-realm operations. New issues might be found in the future. It also does not propose any solution to the issues described in this document. Furthermore, publication of this document does not mean that each of the issues has to be solved by the IETF members. Detailed analysis of the issues, problem definitions, and exploration of possible solutions may be carried out as separate work items.

This document assumes that the readers are familiar with the terms and concepts described in "The Kerberos Network Authentication Service (V5)" ([RFC4120]).

2. Kerberos System

2.1. Kerberos Basic Operation

Kerberos [RFC4120] is a widely deployed authentication system. The authentication process in Kerberos involves principals and a Key Distribution Center (KDC). The principals can be users or services. Each KDC maintains a database of principals and shares a secret key with each registered principal.

The authentication process allows a user to acquire the needed credentials from the KDC. These credentials allow services to authenticate the users before granting them access to the resources. An important part of the credentials is called "tickets". There are two kinds of tickets: the Ticket-Granting Ticket (TGT) and the service ticket. The TGT is obtained periodically from the KDC and has a limited lifetime, after which it expires and the user must renew it. The TGT is used to obtain the other kind of tickets -- service tickets. The user obtains a TGT from the Authentication Service (AS), a logical component of the KDC. The process of obtaining a TGT is referred to as "AS exchange". When a request for a TGT is issued by the user, the AS responds by sending a reply packet containing the credentials, which consist of the TGT along with a random key called the "TGS session key". The TGT contains information encrypted using a secret key associated with a special service, referred to as the "TGS" (Ticket-Granting Service). The TGS session key is encrypted using the user’s key so that the user can obtain the TGS session key only if she knows the secret key she shares with the KDC. The TGT is then used to obtain a service ticket from the TGS -- the second component of the KDC. The process of obtaining service tickets is referred to as "TGS exchange". The request for a service ticket consists of a packet containing a TGT and an "Authenticator". The Authenticator is encrypted using the TGS session key and contains the identity of the user as well as time stamps (for protection against replay attacks). After decrypting the TGT received from the user, the TGS extracts the TGS session key. Using that session key, it decrypts the Authenticator and authenticates the user. Then, the TGS issues the credentials requested by the user. These credentials consist of a service ticket and a session key that will be used to authenticate the user to the desired application service.

2.2. Cross-Realm Operation

The Kerberos protocol provides cross-realm authentication capabilities. This allows users to obtain service tickets to access services in foreign realms. In order to access such services, the users first contact their home KDC asking for a TGT that will be used with the TGS of the foreign realm. If there is a direct trust relationship between the home realm and the foreign realm (practically materialized in shared inter-realm keys), the home KDC delivers the requested TGT.

However, if the home realm does not share inter-realm keys with the foreign realm, we are in a so-called indirect trust model situation. In this situation, the home KDC will provide a TGT that can be used with an intermediary foreign realm that is likely to be sharing inter-realm keys with the target realm. The client can use this "intermediary TGT" to communicate with the intermediary KDC, which will iterate the actions taken by the home KDC. If the intermediary KDC does not share inter-realm keys with the target foreign realm, it will point the user to another intermediary KDC (just as in the first exchange between the user and her home KDC). However, in the other case (when it shares inter-realm keys with the target realm), the intermediary KDC will issue a TGT that can be used with the KDC of the target realm. After obtaining a TGT for the desired foreign realm, the client uses it to obtain service tickets from the TGS of the foreign realm. Finally, the user accesses the service using the service ticket.

When the realms belong to the same institution, a chain of trust can be automatically determined by the client or the KDC by following the DNS domain hierarchy and assuming that a parent domain shares keys with all of its child sub-domains. However, since this assumption is not always true, in many situations, the trust path might have to be specified manually. Since the Kerberos cross-realm operations with the indirect inter-realm trust model rely on intermediary realms, the success of the cross-realm operation completely depends on the realms that are part of the authentication path.

3. Applying Cross-Realm Kerberos in Complex Environments

In order to help the reader understand requirements and restrictions for cross-realm authentication operations, this section describes the scale and operations of two actual systems that could be supported by cross-realm Kerberos. The two systems would be most naturally implemented using different trust models, which will imply different requirements for cross-realm Kerberos.

Hereafter, we will consider an actual petrochemical company [SHELLCHEM], and overview two examples among its plants. Petrochemical companies produce bulk petrochemicals and deliver them to large industrial customers. The company under consideration possesses 43 plants all over the world managed by operation sites in 35 countries. This section shows two examples of these plants.

The first example is a plant deploying a centralized system [CSPC]. CSPC, also referred to as China National Offshore Oil Corporation (CNOOC) and Shell Petrochemicals Company, is operated by a joint enterprise of these two companies. This system is one of the largest of its type in the world. It is located in an area of 3.4 square kilometers in the north coast of Daya Bay, Guangdong, in southeast China. 3,000 network segments are deployed in the system, and 16,000 control devices are connected to local area networks. These devices belong to 9 different subsystems. A control device can have many control and monitoring points. In the plant considered in this example, there are 200,000 control points in all. They are controlled by 3 different control centers.

Another example is a distributed system [NAM]. The Nederlandse Aardolie Maatschappij (NAM) is operated by a partnership company of two enterprises that represent the oil company. This system is composed of some plants that are geographically distributed within the range of 863 square kilometers in the northern part of the Netherlands. 26 plants, each one called a "cluster", are scattered in the area. They are connected to each other by a private ATM WAN. Each cluster has approximately 500-1,000 control devices. These devices are managed by a local control center in each cluster. In the entire system of the NAM, there are one million control points.

In both examples, the end devices are basically connected to a local network by a twisted-pair cable, with a low bandwidth of 32 kbps. End devices use a low clock CPU -- for example, the H8 [RNSS-H8] and M16C [RNSS-M16C]. Furthermore, to reduce power consumption, the clock on the CPU may be lowered. This adjustment restricts the amount of total energy in the device, thereby reducing the risk of explosions.

A device on the network collects data from other devices monitoring the condition of the system. These data are then used to make decisions on how to control other devices via instructions transmitted over the network. If it takes time for data to travel through the network, normal operations cannot be ensured. The travel time of data from a device to another device in both examples must be within 1 second. Other control system applications may have shorter or longer timescales.

Some parts of the operations, such as control, maintenance, and environmental monitoring, can be consigned to an external organization. Also, agents may be consigned to walk around the plant and collect information about plant operations, or to watch the plant from a remote site.

4. Requirements

This section provides a list of requirements derived from the industrial automation use-case. The requirements are written in a generic fashion, and are addressed towards frameworks and architectures that underlie Kerberos cross-realm operations. The aim of these requirements is to provide some foundational guidelines to future developments of cross-realm framework or architecture for Kerberos.

Requirements R-1, R-2, R-3, and R-4 are related to the management of the divided system. Requirements R-5, R-6, and R-7 are related to restrictions in such large-scale industrial networks as those discussed in Section 3.

R-1 For organizational reasons and scalability needs, a management domain typically must be partitioned into two or more sub-domains of management. Therefore, any architecture and implementation solution to the Kerberos cross-realm problem must (i) support the case of cross-realm operations across multiple management domains and (ii) support delegation of management authority from one management domain to another management domain. This must be performed without any decrease in the security level or quality of those cross-realm operations and must not expose Kerberos entities to new types of attacks.

R-2 Any architecture and implementation solution to the Kerberos cross-realm problem must support the co-existence of multiple independent management domains on the same network. Furthermore, it must allow organizations (corresponding to different management domains) to delegate the management of a part of, or the totality of, their system at any one time.

R-3 Any architecture and implementation solution to the Kerberos cross-realm problem must allow the use-case in which one device operationally controls another device, but each belongs to different management domains, respectively.

R-4 Any architecture and implementation solution to the Kerberos cross-realm problem must address the fundamental deployment use-case in which the management domain traverses geographic boundaries and network topological boundaries. In particular, it must address the case where devices are geographically (or topologically) remote, even though they belong to the same management domain.

R-5 Any architecture and implementation solution to the Kerberos cross-realm problem must be aimed at reducing operational and management costs as much as possible.

R-6 Any architecture and implementation solution to the Kerberos cross-realm problem must address the (limited) processing capabilities of devices, and implementations of solutions must be considered to aim at limiting or suppressing power consumption of such devices.

R-7 Any architecture and implementation solution to the Kerberos cross-realm problem must address the possibility of limited availability of communications bandwidth between devices within one domain, and also across domains.

5. Issues

This section lists issues in Kerberos cross-realm operations when used in large-scale systems such as the ones described in Section 3, and taking into consideration the requirements described in Section 4.

5.1. Unreliability of Authentication Chain

When the trust relationship between realms follows a chain or hierarchical model, the cross-realm authentication operations are not dependable, since they strongly depend on intermediary realms that might not be under the same authority. If any of the realms in the authentication path is not available, then the principals of the end realms cannot perform cross-realm operations.

The end-point realms do not have full control of and responsibility for the success of the cross-realm operations, even if their own respective KDCs are fully functional. Dependability of a system decreases if the system relies on uncontrolled components. End-point realms have no way of knowing the authentication result occurring within intermediary realms.

Satisfying requirements R-1 and R-2 will eliminate (or considerably diminish) this issue of the unreliability of the authentication chain.

5.2. Possibility of MITM in the Indirect Trust Model

Every KDC in the authentication path knows the shared secret between the client and the remaining KDCs in the authentication path. This allows a malicious KDC to perform man-in-the-middle (MITM) attacks on communications between the client and any KDC in the remaining authentication chain. A malicious KDC also may learn the service session key that is used to protect the communication between the client and the actual application service. It can then use this key to perform a MITM attack.

In [SPECCROSS], the authors have analyzed the cross-realm operations in Kerberos and provided formal proof of the issue discussed in this section.

Satisfying requirements R-1 and R-2 will eliminate (or considerably diminish) this issue of MITM attacks by intermediate KDCs in the indirect trust model.

5.3. Scalability of the Direct Trust Model

In the direct trust relationship model, the realms involved in the cross-realm operations share keys, and their respective TGS's principals are registered in each other's KDC. Each realm must maintain keys with all foreign realms that it interacts with. This can become a cumbersome task and may increase maintenance costs when the number of realms increases.

Satisfying requirements R-1, R-2, and R-5 will eliminate (or considerably diminish) this issue of scalability of the direct trust model.

5.4. Exposure to DoS Attacks

One of the assumptions made when allowing the cross-realm operation in Kerberos is that users can communicate with KDCs located in remote realms. This practice introduces security threats, because KDCs are open to the public network. Administrators may think of restricting the access to the KDC to the trusted realms only. However, this approach is not scalable and does not really protect the KDC. Indeed, when the remote realms have several IP prefixes (e.g., control centers or outsourcing companies, located worldwide), then the administrator of the local KDC must collect the list of prefixes that belong to these organizations. The filtering rules must then explicitly allow the incoming traffic from any host that belongs to one of these prefixes. This makes the administrator's tasks more complicated and prone to human errors. Also, the maintenance cost increases. On the other hand, when a range of external IP addresses are allowed to communicate with the KDC, then the risk of becoming targets of attacks from remote malicious users increases.

Satisfying requirements R-1, R-3, R-4, and R-5 will eliminate (or considerably diminish) this issue of exposure to denial-of-service (DoS) attacks.

5.5. Client’s Performance

In Kerberos cross-realm operations, clients have to perform TGS exchanges with all of the KDCs in the trust path, including the home KDC and the target KDC. A TGS exchange requires cryptographic operations and may consume a large amount of processing time, especially when the client has limited computational capabilities. As a result, the overhead of Kerberos cross-realm exchanges may cause unacceptable delays in processing.

We ported the MIT Kerberos library (version 1.2.4), implemented a Kerberos client on our original board with H8 (16-bit, 20 MHz), and measured the process time of each Kerberos message [KRBIMPL]. It takes 195 milliseconds to perform a TGS exchange with the on-board H/W crypto engine. Indeed, this result seems reasonable, given the requirement of the response time for the control network. However, we did not modify the clock speed of the H8 during our measurement. The processing time must be slower in an actual environment, because H8 is used with a lowered clock speed in such a system. With such devices, the delays can become unacceptable when the number of intermediary realms increases.

Satisfying requirements R-1, R-2, R-6, and R-7 will eliminate (or considerably diminish) this issue relating to the client’s performance.

5.6. Kerberos Pre-Authentication Problem in Roaming Scenarios

In roaming scenarios, the client needs to contact her home KDC to obtain a cross-realm TGT for the local (or visited) realm. However, the policy of the network access providers or the gateway in the local network usually does not allow clients to communicate with hosts in the Internet unless they provide valid authentication credentials. In this manner, the client encounters a chicken-and-egg problem where two resources are interdependent; the Internet connection is needed to contact the home KDC and for obtaining credentials, and on the other hand, the Internet connection is only granted for clients who have valid credentials. As a result, the Kerberos protocol cannot be used as it is for authenticating roaming clients requesting network access. Typically, a VPN approach is applied to solve this problem. However, we cannot always establish VPNs between different sites.

Satisfying requirements R-3, R-4, and R-5 will eliminate (or considerably diminish) this roaming-related issue pertaining to Kerberos pre-authentication.

6. Implementation Considerations

This document describes issues of Kerberos cross-realm operations. There are important matters to be considered when designing and implementing solutions for these issues. Such solutions must not introduce new problems. Any solution should use existing components or protocols as much as possible, and it should avoid introducing definitions of new components. It should not require new changes to existing deployed clients and as much as possible should not influence the client code-base. Because a KDC is a significant server in an information system based on Kerberos, any new burden placed on the KDC should be minimal. Solutions must take these tradeoffs and requirements into consideration. On the other hand, solutions are not required to solve all of the issues listed in this document at once.

7. Security Considerations

This document clarifies the issues of the cross-realm operation of the Kerberos V system, which include security issues to be considered. See Sections 5.1, 5.2, 5.3, and 5.4 for further details.

8. Acknowledgements

The authors are grateful to Nobuo Okabe, Kazunori Miyazawa, and Atsushi Inoue. They gave us lots of comments and suggestions for this document from its earliest stages. Nicolas Williams, Chaskiel Grundman, and Love Hornquist Astrand gave valuable suggestions and corrections. Thomas Hardjono devoted much work and helped to improve this document. Finally, the authors thank Jeffrey Hutzelman. He gave us a lot of suggestions for completion of this document.

9. References

9.1. Normative References

[RFC4120] Neuman, C., Yu, T., Hartman, S., and K. Raeburn, "The Kerberos Network Authentication Service (V5)", RFC 4120, July 2005.

9.2. Informative References

[CSPC] "CSPC Nanhai - Shell Global Solutions", <http://www.shell.com/home/content/global_solutions/aboutshell/key_projects/nanhai/>.

[KRBIMPL]   "A Prototype of a Secure Autonomous Bootstrap Mechanism for Control Networks", Nobuo Okabe, Shoichi Sakane, Masahiro Ishiyama, Atsushi Inoue and Hiroshi Esaki, SAINT, pp. 56-62, IEEE Computer Society, 2006.

[NAM]   Nederlandse Aardolie Maatschappij BV, <http://www.nam.nl/>.

[RNSS-H8]   "H8 Family | Renesas Electronics", <http://www.renesas.com/products/mpumcu/h8/h8_landing.jsp>.

[RNSS-M16C] "M16C Family (R32C/M32C/M16C) | Renesas Electronics", <http://www.renesas.com/products/mpumcu/m16c/m16c_landing.jsp>.

[SHELLCHEM] "Shell Chemicals", <http://www.shell.com/home/content/chemicals>.

[SPECCROSS] I. Cervesato and A. Jaggard and A. Scedrov and C. Walstad, "Specifying Kerberos 5 Cross-Realm Authentication", Fifth Workshop on Issues in the Theory of Security, January 2005.

Authors' Addresses

Shoichi Sakane
Yokogawa Electric Corporation
2-9-32 Nakacho, Musashino-shi
Tokyo 180-8750 Japan
EMail: Shouichi.Sakane@jp.yokogawa.com

Ken'ichi Kamada
Yokogawa Electric Corporation
2-9-32 Nakacho, Musashino-shi
Tokyo 180-8750 Japan
EMail: Ken-ichi.Kamada@jp.yokogawa.com

Saber Zrelli
Yokogawa Electric Corporation
2-9-32 Nakacho, Musashino-shi
Tokyo 180-8750 Japan
EMail: Saber.Zrelli@jp.yokogawa.com

Masahiro Ishiyama
Toshiba Corporation
1, Komukai Toshiba-cho, Saiwai-ku
Kawasaki 212-8582 Japan
EMail: masahiro@isl.rdc.toshiba.co.jp