~~ANL/ECT/CP--90110~~
CONF-9605174--1
ANL/ECT/CP--90110

An Authentication Infrastructure for Today and Tomorrow

Douglas E. Engert

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

The submitted manuscript has been authored by a contractor of the U. S. Government under contract No. W-31-109-ENG-38. Accordingly, the U. S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U. S. Government purposes.

Argonne National Laboratory
Electronics and Computing Technologies Division
Operated by The University of Chicago for the U. S. Department of Energy under Contract W-31-109-Eng-38

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

MASTER

The Open Software Foundation's Distributed Computing Environment (OSF/DCE) was originally designed to provide a secure environment for distributed applications. By combining it with Kerberos Version 5 from MIT, it can be extended to provide network security as well. This combination can be used to build both an inter and intra organizational infrastructure while providing single sign-on for the user with overall improved security. The ESnet community of DOE is building just such an infrastructure. ESnet has modified these systems to improve their interoperability, while encouraging the developers to incorporate these changes and work more closely together to continue to improve the interoperability. The success of this infrastructure depends on its flexibility to meet the needs of many applications and network security requirements. The open nature of Kerberos, combined with the vendor support of OSF/DCE, provides the infrastructure for today and tomorrow.

INTRODUCTION

ESnet is a nationwide computer data communications network managed and funded by the U.S. Department of Energy Office of Energy Research (DOE/OER). ESnet interconnects the diverse and widely dispersed energy research community which consists of the Department of Energy national laboratories and U.S. universities funded by the DOE to conduct energy research.

Two of the ESnet committees are the ESnet Site Coordinating Committee (ESCC) and Distributed Computing Coordinating Committee (DCCC). The ESCC is a standing committee which acts as an advisory body, providing a forum for the consideration of a broad range of technical issues and interchange about ESnet-wide activities. DCCC actively works to facilitate the development of a functional distributed computing environment for the sciences associated with DOE research.

The Authentication Task Force is a joint task force between the ESCC and the DCCC. It has recommended setting up an authentication infrastructure based on Kerberos Version 5 and the Open Software Foundation's Distributed Computing Environment (OSF/DCE). Through the Distributed Informatics, Computing & Collaborative Environment (DICCE) project, four of the National Laboratories, Argonne National Laboratory (ANL), The National Energy Research Scientific Computing (NERSC) located at Lawrence Berkeley National Laboratory (LBNL), Pacific Northwest National Laboratory (PNNL) and Sandia National Laboratory (SNL) have been funded to implement this technology at their local sites.

More information about ESnet and the above activities, including the Authentication Infrastructure of the DICCE proposal can be found at: "http://www.es.net"

INFRASTRUCTURE

One of the aspects of distributed computing which is often overlooked is user authentication. Many user's have to use multiple distributed computing environments, which may be located at different Laboratories. Many of a Laboratories user's are off site at educational institutions.

These user's are subjected to multiple policies and procedures, including multiple accounts with multiple passwords. In some cases Smart cards are required, in most only standard password access is used.

Not only does the use of multiple accounts lead to poor password management on the user's behalf, but it is difficult to audit, passwords are easily stolen and/or shared, and there is no association between different user identities at different organizations.

The concept of single sign-on, where the user authenticates once and this authentication is accepted across a wide area network such as ESnet would solve many of the above problems, as well as simplifying the user's environment.

Building a common authentication infrastructure would also allow for the introduction of smart cards or other similar devices. It would be the foundation for mutual authentication and allow for delegation of authority.

KERBEROS VERSION 5 - BASIS FOR AUTHENTICATION

The Kerberos Network Authentication Service (V5) as documented in RFC 1510, provides a means of verifying the identities of principals on an open network. Kerberos was originally developed as part of the Project Athena at MIT. Since then it has undergone a number of revisions, and Kerberos V5 is the latest. MIT has continued its development and has made the source available, as beta level code, subject to export restrictions.

The availability of the source has made this attractive to many in the Internet community who have assisted in its development by reporting problems, and suggestions. MIT personnel have been responsive to this feedback and have incorporated many of the suggestions. The openness of the process and the availability of the source has also lead to careful inspection of the code for possible security problems. Thus far, it has proven to be a reliable and secure authentication service.

A number of commercial security systems have been based on the Kerberos specification, or the MIT code directly. These have mostly been smaller companies. Many of their products interoperate, but not all.

Any solution for a common infrastructure requires interoperability, and must support authentication to a wide range of applications and platforms. Kerberos V5 meets these needs.

OSF/DCE - USES KERBEROS VERSION 5 FOR AUTHENTICATION

The Open Software Foundation is a not-for-profit, membership-based organization chartered to assist the information technology industry in research, development, and delivery of key vendor-neutral technology (software source code) to make open systems possible. Its Distributed Computing Environment (DCE) is an integrated set of services to support distributed applications. It uses the Kerberos Version 5 protocol as documented in RFC 1510 for authentication. This makes DCE attractive from a security view point.

Because DCE is based on Kerberos V5, its security server can serve both DCE and Kerberos V5 clients. This has been demonstrated by the DICCE project members. This makes DCE attractive as the bases for a common authentication service.

Almost every major computer vendor is a member of OSF, and DCE is commercially available for almost every computer platform. Many of the DCE client services are now being bundled with the vendors' base operating systems. This acceptance of DCE by the vendors makes DCE attractive from a maintenance viewpoint.

DCE is also being accepted by many of the application vendors, including many of the database vendors, i.e. Oracle and Sybase. Its secure RPC and threads provides vendors with a common technology upon which to build their distributed applications. This makes DCE attractive from the integrated business application view point.

Not only does DCE provide a Security server for authentication, it provides a Privilege Server for authorization. The Privilege Server issues Privilege Attribute Certificates (PAC) which contain, among other things, the list of groups of which the user is a member. This information can then be used by an applications to check against Access Control Lists (ACL).

MISSING FUNCTIONALITY OF OSF/DCE

DCE was originally designed to support distributed applications. It relies on the security of the vendor's operating system upon which it runs. The security of the traditional network applications such as telnet, FTP, POP, the Berkeley "r" commands, rlogin, rsh, rcp, etc. were not considered to be supported by DCE, but rather part of the underlying vendor's operating system.

Many of today's security problems are related to the security problems in these traditional network applications. These are the applications the system manager uses to maintain the systems upon which DCE is based. Thus the DCE security depends on security of these traditional network applications.

These are the same applications which Kerberos first addressed as a Network Authentication Service. Versions of the Berkeley "r" commands, telnet, FTP, and POP are readily available, with the MIT Kerberos source and as commercial products.

By combining the best of DCE and Kerberos V5, comprehensive Application and Network security can be obtained. The Authentication Task Force showed that it was feasible to use the DCE security server as the Kerberos KDC. The DICCE Project is continuing that work by working with the providers to continue to improve the interoperability of DCE and Kerberos V5. Many suggestions and fixes have been provided to MIT and to OSF.

The OSF/RFC 92.0 "DCE Interoperability with Kerberos - Functional Specification" addresses many but not all of these applications. It promises security server and Berkeley rlogin and rsh commands and daemons which will interoperate with MIT Kerberos Version 5 beta release 4 or 5.

OTHER APPLICATIONS

Initially not every client workstation will have DCE. Many will only have Kerberos V5. A user may wish to login to a DCE client and obtain a DCE context. This is needed to access DFS. This context is normally obtained by providing the userid and password, but it can also be obtained by the use of a Kerberos V5 ticket. The DICCE Project members used a little known feature in DCE to do this, and by making changes to the Kerberos daemons and using forwarded Kerberos V5 tickets, a DCE context was obtained automatically during login by the k5dcelogin program. The OSF/RFC 92.0 refers to this work and the method used by the DICCE Project.

DCE was released with one major application, the Distributed File System (DFS) and relies on DCE services. Among its many features is the ability to protect individual files with Access Control Lists. It is much more robust and complicated than its predecessor, the Andrew File System (AFS).

Many of the ESnet organizations currently use AFS and intend to continue to use it as they learn more about DFS. AFS is the product of the Transarc Corporation who developed DFS for OSF. AFS is loosely based on Kerberos V4 for authentication, and uses the concept of a token. The token is presented to the file server with each request. In reality, the token is the encrypted section of a Kerberos V4 ticket which is stored in the system kernel.

The DICCE project members modified the aklog program to have it work with Kerberos V5. Originally aklog would obtain a Kerberos V4 ticket and convert it to an AFS token. The modified version obtains a Kerberos V5 ticket, converts it to a Kerberos V4 ticket, and then to an AFS token. This is done by using the krb524 routines and the krb524d daemon which are part of the MIT Kerberos distribution. The token can be obtained automatically during login by the k5afslogin program, using a forwarded Kerberos V5 ticket. This can be combined with the k5dcelogin program to get both a DCE context and an AFS token from the same forwarded ticket. The k5afslogin program is independent and can run on DCE and non-DCE clients.

Authenticated World Wide Web access is another important application which can be addressed by DCE and Kerberos V5. DICCE project members have experimented with the Kerberized version of the NCSA Web server and browser. It can use Kerberos V4, AFS, or Kerberos V5 for authentication. We tested it using the DCE security server with the Kerberos V5 authentication including cross-realm authentication. We demonstrated this at the fall ESnet meeting, however it requires the use of the NCSA Mosaic browser. We have many people who would like to use other browsers.

Other DICCE project members have experimented with the OSF Research Institute's DCE server and Browser proxy. This allows the user to use his favorite browser, yet gives full DCE security including privacy of the data transmitted across the network. This has just been released by Gradient Technologies as the WebCrusader product family and looks very promising.

The High Performance Storage System (HPSS) being developed at the National Storage Laboratory (NSL) at LLNL is based on DCE. Having adopted DCE as the system infrastructure, HPSS can leverage other DCE or DCE based services in the future.

POINTS OF INTERACTION

As implied above, the DCE security server can issue Kerberos tickets, and act as a Kerberos V5 KDC. It can not act as a Kerberos V4 KDC. The MIT KDC can issue both Kerberos V5 and V4 tickets. The MIT KDC can not act as a DCE security server since it does not respond to RPCs and does not issue Privilege Authorization Certificates (PAC) which are used by DCE for authorization. Thus you must use the DCE security server in any mixed environment. This presents a problem for those organizations who have a large base of Kerberos V4 applications or users. There is no clear cut strategy for these sites to migrate to DCE directly.

Both the MIT and DCE developers are adhering to the Kerberos RFC 1510. But the RFC only covers the network protocol. When DCE and Kerberos clients are both operating on the same machine, there are files which need to be shared and other points of contention.

The MIT development work of Kerberos has continued as has the development of DCE. Both groups have made changes and extensions in areas which could cause interoperability problems, being careful to preserve much of the interoperability with the others products.

Kerberos tickets are stored in a ticket cache which is pointed at by the KRB5CCNAME environment variable. This is usually a file which can be shared by both Kerberos and DCE clients. MIT has added additional fields, as compared to the DCE version, but MIT has also added backward compatibility options. DCE added additional information in opaque fields of the tickets which are not used by Kerberos. DCE also added additional files which are related to the ticket cache.

This means that if DCE clients are to use this file, the dce_login program must be used to initialize it, and get the initial Kerberos V5 Ticket Granting Ticket (TGT). Kerberos clients can then use the ticket cache, to store additional tickets. The DCE kinit program must be used with the DCE created cache. Either the DCE or Kerberos klist commands can be used, each lists its own sections of the file. The DCE kdestroy should be used, as it destroys the related files as well as the ticket cache. The Kerberos kinit, klist, and kdestroy can be used, but DCE clients can not use this cache, as the related PAC information is not available.

Keys for application and daemon processes are stored in a keyfile. These can also be shared, and the DCE commands such as the ktadd subcommand of the DCE rgy_edit command can be used to update it. It is interesting to note that the default location of the systems keytab file for DCE is "/krb5/v5srvtab", while the Kerberos default is "/etc/v5srvtab". This goes back to the early roots of the Kerberos code in DCE. By adding a symbolic link from /etc to /krb5 the systems can share the same file.

Kerberos uses a configuration file on the client to find the location of the KDC. Early versions of Kerberos and DCE used the /krb5/krb.conf file. Newer versions of Kerberos use the

/krb5/krb5.conf which has a different format and contains additional information. The system administrator must keep these files in synchronization.

Applications can not be linked with both the DCE libraries and the Kerberos libraries. Many of the routines have the same name but have different parameters. The internal structures have also changed. The API for the version of Kerberos used by DCE is not exposed. The Kerberos header files are not provided with DCE which presented major problems when converting a Kerberos forwarded ticket to a DCE context. These problems were solved by having rlogind "exec" the k5dcelogin, which then "exec"s the login progra; rlogind is linked with the Kerberos libraries, k5dcelogin is linked with the DCE libraries.

The problem of using a forwarded Kerberos ticket to get a DCE context in the ftpd module is still unsolved. The "exec" trick can not be used. Using dynamically loaded routines may solve this problem.

CROSS REALM/CELL CONSIDERATIONS

One of the improvements in the Kerberos V5 protocol over the Kerberos V4 protocol is the ability to do Cross-Realm authentication using intermediate realms. This is accomplished by having two realms exchange inter-realm keys. In DCE this is done with the rgy_edit cell command. A client can then use its TGT to request a TGT for another realm. The client can then repeat the process using the new TGT to request another TGT for a third realm, etc. Kerberos V5 adds the transited field to the ticket which is updated by each realm as it grants one of these tickets. The end service, or intermediate KDC can check this transited field and reject the request if it does not trust any of the realms involved in the authentication. Kerberos V4 did not have the transited field, but did allow for direct connections, i.e. n(n-1)/2 inter-realm keys need to be exchanged.

RFC 1510 states: "Realms are typically organized hierarchically." It implies that this hierarchy is based on the realm name. It goes on to say a database could be used to construct an authentication path. The ESnet Authentication Pilot Project demonstrated the use of a database to construct the authentication path. This removed the restrictions imposed on the choice of realm names.

A version of the Configurable Authentication Path code has been included in the MIT source since December of 1995. The "database" which defines the authentication path is a section in the /krb5/krb5.conf file.

Early versions of Kerberos V5 and DCE did not completely implement cross-realm, and DCE 1.1 still does not. DCE 1.0.3a did not update or check the transited field. This is a security problem, if a number of realms/cells are sharing keys who do not fully trust each other.

OSF had proposed a different database and a different hierarchical structure. OSF has not included the Configurable Authentication Path code yet. For interoperability, both DCE and other Kerberos implementations need to agree on similar methods of a client determining the authentication path, and the server checking the authentication path. At the minimum this may

be done by the system administrators maintaining two separate databases with similar information.

The use of a database does not scale well, as does the need to have many intermediate realms. A better approach would be to use some public key method for the exchange of the inter-realm keys. But since the client, server, and KDC are all involved, to insure interoperability, this implementation needs to be a coordinated effort among all the Kerberos vendors.

FUTURES

Kerberos defines a user-to-user protocol, which allows a user session to act as a server. OSF has announced the intent to support this protocol in DCE. The use of this feature with end user workstations such as X terminals or personal computers could greatly simplify the administration of these workstations. Currently, the workstation needs a principal and a key which requires additional system administration.

X11R6 has some Kerberos V5 capabilities which have not been well tested with the newer versions of Kerberos. This includes the capability to use the user-to-user authentication. By using the end users session key for the X-server, the X-terminal does not need to have a principal. This makes it an attractive terminal for public areas, including conferences and shows where users could use them to authenticate to their home cell.

There are a number of terminal servers on the market today which support Kerberos, but they require the user to send his userid and password to the terminal server over the phone lines. This exposes the password and is not in the spirit of single sign-on. One way to address the problem is by having the terminal server set up the SLIP or PPP connect, but act as a selective router, only allowing connections to the Kerberos KDC or DCE security server and possibly DNS servers. This allows the user to authenticate and get a ticket for the terminal server. When authenticated to the terminal server, it would then allow full access to the network.

The ability to do true single sign-on at the workstation, i.e. not sending passwords over the network, and only having to enter the password once, sets the stage for the wide spread use of smart cards or other devices. To use these effectively, you need to have the infrastructure in place, which can be the same infrastructure provided by DCE and Kerberos. Having cross-cell authentication in place, means that a user only needs a single smart-card. Many organizations are standardizing on employee badges which could be a combination smart card and badge. The combination could cut overall costs and improve security.

CONCLUSION

OSF/DCE provides the authentication and authorization infrastructure for many applications, DFS being just one of them. However the learning curve, the cost of implementation, and the current lack of applications, combined with the perception of little benefit by the end user, has hindered its widespread implementation.

Only by utilizing currently Kerberized applications such as telnet, ftp, POP, and "r" commands and adding additional applications such as AFS, can we start to improve the knowledge base and perception of the users and administrators, and build the infrastructure which can be used by many other applications.

This approach will succeed and mature, not because of DFS or any other single application, but because it provides the open, unified infrastructure upon which to build.