Open Software Foundation
Request For Comments: 63.1

R. Salz (OSF)

May 1995

DCE 1.2 CONTENTS OVERVIEW

1. INTRODUCTION

This document describes the goals and contents of DCE 1.2. The purpose of the document is to inform the reader of the features of the project and solicit comments on it. This document serves as part of the continuing process of factoring membership input into the development of DCE.

DCE 1.2 is the first project to be operated under OSF's pre-structured technology (PST) model. In this model the project is planned, managed, developed, and funded by a group of project sponsors. The sponsors of the DCE 1.2 PST are Digital, HP, Hitachi, and IBM. These sponsors, a representative of the OSF End User Steering Committee, and a representative of the OSF permanent staff form a Project Steering Committee (PSC) which is responsible for running the program. For more information on the PST process, see [RFC 62].

Within the PST model, DCE 1.2 will introduce the concept of a ''sub-PST''. In this PST, the sub-PST's are viewed as sub-projects under the overall DCE 1.2 ''umbrella''. This distinction is primarily one of project management. OSF will be responsible for general oversight and coordination of all sub-PST's.

This document is a product of the DCE 1.2 PSC. It is derived from information provided by the DCE Technical Planning Committee (TPC), which is comprised of lead architects from each of the sponsoring companies and representatives from the OSF. The TPC also serves as a continuing architectural review body for DCE.

The contents of the project were determined by the PSC after gathering information from a variety of sources: OSF SIG's, OSF staff, the OSF End User Steering Committee, as well as each sponsor company's marketing and engineering departments. The project described here meets the technical requirements gathered by the sponsors, can be delivered in two releases approximately 12 and 18 months after the delivery of DCE 1.1, and can be completed within the budget set by the sponsors.

1.1. OSF Member Feedback

Comments regarding this document should be sent to the `dce-1.2-comments@osf.org` mailing list and to the appropriate OSF SIG's (e.g., `sig-dce@osf.org`, `sig-security@osf.org`).

Functional specifications of the technology deliverables will be published as OSF RFC's; many are already available and are noted below. See [RFC Index] for details.

1.2. Document Structure

The remainder of this document is divided into four sections. Section two describes the major goal and motivation of the project. Section three describes the technical contents of the project. Section four describes the current state of the project and the planned contents of each release. Section four also enumerates and explains the changes between this document and the previous version.

This document represents the efforts of the DCE 1.2 Project Steering Committee, the DCE Technical Planning Committee, and staff at the sponsor companies and OSF. There are too many individuals to name them all, but ''you know who you are''.

2. THE MAJOR GOAL OF DCE 1.2: EXPANDING DEPLOYMENT

The primary goal for DCE 1.2 is to facilitate the continuing trend toward enterprise-wide deployment of DCE in the 1996–1997 timeframe. To reach this goal we need to continue to make DCE more scalable, easier to program, more easily manageable, and more smoothly integrated with the pre-existing distributed computing technologies that exist alongside DCE in many enterprises.

DCE 1.2 builds upon a series of releases that provide a strong basis for establishing DCE as an enterprise-wide distributed computing solution. Progress in the areas of performance, robustness, scalability, richness of programming tools, and ease of administration have made DCE capable of serving as the infrastructure upon which businesses can build a coherent distributed computing environment using heterogeneous systems. DCE 1.2 will continue on that path.

With DCE's increasing deployment, the requirements for improvement have become more focused, reflecting actual experience rather than speculative interest. Users and administrators have identified where improvements are needed, and DCE 1.2 will address those areas.

DCE 1.2 focuses on broadening the appeal of DCE rather than on extending the technology for the sake of technology development. It will be a balance of new user-oriented features and basic capabilities which set the stage for future functionality. As DCE matures, the trend will be one of evolution rather than revolution. DCE 1.2 will protect the user's investment in DCE while continuing to move forward.

3. TECHNOLOGY DESCRIPTIONS

The technology in this PST can be divided into four major areas: integration with other environments, ease of administration, ease of programming, and scalability.

Each section below starts with an overview of the area. Each overview contains a short summary of the issues being addressed and usually contains a set of one-sentence descriptions of how DCE 1.2 will address them. Following the summary, there are more detailed sub-sections explaining each technology deliverable.

3.1. Integration with Other Environments

Few DCE installations are deployed in isolation: most environments include a mixture of existing technologies that have some degree of overlap with services provided by DCE. DCE 1.2 will address this by allowing access between DCE and the most popular of these environments.

The desktop is becoming the paramount platform in the enterprise. Currently Novell's Netware is the dominant network operating system. Sun's Open Network Computing system is present on many enterprise platforms as well. Therefore, DCE 1.2 provides features to help integrate Netware and ONC with DCE facilities.

MIT's Kerberos network authentication environment has gained support in the commercial sector. More and more, users demand that the Kerberos environment provided as part of DCE be compatible with the standard version defined by the IETF. DCE 1.2 will guarantee this wire protocol compatibility.

Public key technology is becoming an increasingly important mechanism in many computing environments and applications. Many enterprises need login technology that does not require private keys. In addition, tools to support use of public key certificates will allow DCE to integrate smoothly into the evolving public key oriented computing environment.

The DCE security runtime supports the RPC client-server programming model. DCE 1.2 will provide peer-to-peer authentication. This allows other communication models such as message-oriented middleware to use the DCE security infrastructure.

3.1.1. Netware coexistence

Novell Netware has established itself as the dominant information sharing technology in small personal computing networks. To establish a coherent enterprise-wide computing environment, the large population of Netware users must be integrated into the DCE environment. DCE 1.2 addresses this integration by providing file sharing services and administrative aids that allow Netware 3.X users and DCE users to have a single identity and access to the DCE filesystem, DFS.

In DCE 1.2, Netware clients can define any directory tree in DFS as a Netware volume and use standard Netware or DOS/Windows commands and API's to manipulate the files and directories under it. File access security is maintained through a gateway similar to the NFS gateway delivered as part of DCE 1.1. By adding data to the NetWare user object, DCE 1.2 can map a Netware user into an authenticated DCE principal. The gateway will map identities, filenames, and permissions across the two domains.

3.1.2. ONC coexistence

ONC, especially NFS, exists on a large number of machines in many different operating environments. Both DCE and ONC provide mechanisms for managing users in the distributed environment. DCE 1.2 will reduce the administrative burden of running DCE and ONC services in the same environment by providing mechanisms to enable synchronization between the ONC NIS password and group maps with the corresponding information found in the DCE registry. This work is being done in two parts:

(a) A new class of security replicas will be provided, allowing gateway servers to capture updates to the registry and propagate them out into other domains.

(b) The DCE security server will be able to store a user's secret key in plaintext.

The secure NFS protocol gateway planned for DCE 1.2 but accelerated into the DCE 1.1 release makes NFS client machines instant consumers of the DCE distributed file system. Additional support for DFS host-specific and architecture-specific (`@HOST` and `@SYS`) file naming features will be added.

3.1.3. Kerberos V5 support

The DCE security service includes an implementation of the MIT Kerberos Version 5 (V5) authentication and key distribution service. Prior to DCE 1.2 there has been no interoperability commitment in the OSF DCE offering.

The Kerberos V5 protocol has become more stable with the release of IETF-RFC 1510 [Kohl] and its movement through the IETF standards process. DCE 1.2 will enhance the high degree of interoperability that existed in previous releases with committed support for the IETF-RFC 1510 protocol. This support will formally allow Kerberos V5 applications running on either DCE or non-DCE platforms to access the DCE security server as a full-function IETF-RFC 1510 Kerberos server. Prior to the release of DCE 1.2, a specific MIT release will be established as the interoperability base.

Many consumers of DCE wish to extend the single login environment. The MIT Kerberos environment has traditionally included network utilities that transmit user account information (e.g., `telnet`, `rlogin`, and `ftp`). These utilities are integrated with Kerberos to achieve a single login facility in the networked environment. Earlier DCE releases have not included these utilities, in part because IETF standardization has not been completed. DCE 1.2, however, will include implementations of `telnet`, `rlogin` and `ftp` that will use DCE's Kerberos facilities to avoid exposing passwords on a network. Any modifications to these protocols not yet completed by the IETF will be made available to the IETF by the DCE project team for potential inclusion in the IETF standards.

3.1.4. Public key support

DCE 1.2 will allow public key technology to be used to support login. With this technology, the security server need not store the long term key (or password) for a principal so that it will remain undisclosed should any compromise of the security server occur. The system will allow some principals to use the pre-DCE 1.2 mechanisms while others have access to the public key mechanism; it will retain full interoperability with existing DCE releases.

At login, public key users will receive credentials that allow them to use the current Kerberos-based DCE authentication mechanism. The DCE 1.1 pre-authentication protocol [RFC 26] is used so that the login client need not determine whether a given user is public-key-capable prior to requesting credentials. In addition to login, the password-change protocol will be enhanced to allow users to change either their password or their private-key.

For details, see [RFC 68] and [RFC 80].

3.1.5. User-to-user authentication

When the concept of ''server'' is associated with a long running system resource — like the name server or the security server — it seems natural that the server is running on a machine with access to the long term key to the identity of that server. If for no other reason, the server is not normally associated with a human user but rather with a pseudo-user corresponding to the system service. This does not, however, map well onto all application domains. In particular, some applications need a ''peer-to-peer'' model, such as is described as requirement D3 of [RFC 8].

The user-to-user authentication facility will provide an alternate Ticket Granting Service (TGS) protocol as defined in IETF-RFC 1510. This will provide server applications with the same sort of insulation from a principal's long term key that is available for client applications. In particular it will be possible to direct a protected RPC to a program that only has a login context, and no key table (file) or other access to a long-term key.

3.2. Ease of Administration

DCE 1.2 will enhance inter-cell cooperation by allowing groups to contain foreign principals.

3.2.1. Multi-cell groups

DCE 1.2 will allow principals from a foreign cell to be added to a group in the local cell. For example, this will allow cross-cell cooperation among DCE time services. It should also ease enterprise-wide security administration, cell reconfiguration, and other management tasks.

3.3. Ease of Programming

Application development will be simplified through the support for C++ environments and single-threaded client applications.

3.3.1. IDL C++ support

DCE 1.2 will add C++ support to IDL. This support allows client and server programs written in C++ to utilize DCE RPC in a highly transparent manner using natural C++ constructs. The IDL language will be extended to support C++ features such as inheritance and object references.

A client using these features can create remote objects or look up existing remote objects by invoking the static member functions of the interface. Once an object reference is obtained, any of its member functions can be invoked. Local and remote objects can both be used in the application with no client code differences.

The IDL C++ support provides language enhancements and related runtime functions for applications to manage distributed objects. Application developers can use the C++ support as is, or as the underlying framework for an object model at a higher level of abstraction. The C++ support does not force adoption of any one object model or class hierarchy, providing the developer with much flexibility.

For more details, see [RFC 48].

3.3.2. Single-threaded RPC

Many existing applications that would benefit from deployment in a distributed environment have not been written to be thread-aware and do not run properly in a threaded environment. In addition, developers often use third-party packages, which are themselves not thread-aware. The existence of a thread-free version of DCE RPC would increase software reuse by making it substantially easier for these applications to be adapted to the DCE.

In DCE 1.2 some RPC client applications will no longer have a dependency on a threaded environment. This simplifies the task of application developers, eliminating the need for thread-aware programmers, debuggers and support libraries.

Some of the techniques for achieving this work item have been described in [RFC 31]. The client side of the RPC NSI has also had its dependency on threads removed.

3.4. Scalability

DCE 1.2 is addressing implementation limits that restrict scaling. While DCE 1.2 is considering the long term implications of scaling to cells with millions of users and hosts, short term activities are also being pursued to make sure that cells scale well when dealing with 100,000 users, hosts, or servers.

3.4.1. Improvements to CDS

The changes to CDS address the robustness of the CDS server. These changes will significantly reduce the use of virtual memory at a server that is receiving many updates. The changes for DCE 1.2 will include replacing the data storage backend with a more efficient one, reducing the duplication of information at a server and more aggressively purging stale data.

DCE 1.2 will also supply a salvager utility that will replace current unsupported tools. The utility will resemble the interface and function of the security salvager: it will dump the database to a file, check for and fix detectable corruption, and allow an administrator to explicitly remove data.

For details, see [RFC 69].

3.4.2. Improvements to security

Simple changes to the security server will be made to obtain considerable performance improvements when dealing with large cells (those with more than 50,000 principals). This includes documenting the configurable checkpoint interval and partitioning internal datasets so that the amount of data written to disk during a checkpoint is proportional to the amount of data modified.

In addition, DCE 1.2 will identify bottlenecks and places where resource consumption could be reduced and address them as possible.

3.4.3. Improvements to DFS

There are a number of areas where DFS performance will be improved and made significantly more stable.

3.4.3.1. Optimized token manager

Improvements to the token manager will decrease the memory requirements and improve performance and reliability of DFS. The resulting token manager should be about half the size and offer a factor of six improvement in performance of token-related operations.

For details, see [RFC 73].

3.4.3.2. DFS server preferences

Currently DFS makes an arbitrary choice of a file server which could cause poor performance and reduce the ability to scale in a WAN environment. DCE 1.2 will allow an administrator to identify server preferences on a per-fileset basis. Default preferences will be based on IP subnet numbers. This feature will allow DFS clients to make intelligent choices about which servers to use for different filesets.

For details, see [RFC 74].

3.4.3.3. Vnode/VM management

DCE 1.2 will improve the way that LFS handles vnodes. These improvements will allow DFS to perform significantly better as the system is subjected to higher levels of stress. In particular, finer-grain locking will operate more efficiently at high, concurrent administrative and user loads, and will operate more efficiently on multiprocessor machines. In addition, an improved interface between LFS and the native virtual memory subsystem should simplify I/O access patterns and code paths.

For details, see [RFC 75] and [RFC 78].

3.4.3.4. Replication enhancements

A fundamental principle of DCE is that services should be replicated for high availability and enhanced scalability. The DFS architecture provides replicated filesets as a way to implement this.

Planned enhancements improve the DFS replication implementation to achieve greater reliability and better performance. Users will get the benefit of a system which has higher availability and better scalability. Administrators will be able to reorganize how data are located and replicated.

DCE 1.2 will convert the replication server from a serial server to a concurrent server. The existing DFS replication server accepts updates to replicas serially: if several replicated filesets are being retrieved from a collection of servers, the changes from the separate (read/write) filesets are retrieved one after the other. In a large system where many filesets are being propagated, this algorithm represents a bottleneck to scalability.

For details, see [RFC 76].

3.4.3.5. Multi-home support

DFS currently requires all clients and servers to be reachable via all network interfaces. DCE 1.2 will enhance DFS servers to perform better on hosts connected through multiple interfaces to multiple networks.

4. PROJECT STATUS AND RELEASE CONTENTS

The DCE 1.2 PST has been approved by the OSF Board and the project is now in the development and funding phases, as defined by the PST process.

4.1. Release Contents

As stated above, the DCE 1.2 PST will be made available in two releases: DCE 1.2.1 in November 1995, and DCE 1.2.2 in June 1996. The contents are as follows:

(a) DCE 1.2.1:
- (i) NFS gateway.
- (ii) Security replication.
- (iii) IDL C++ support.
- (iv) Optimized token manager.
- (v) DFS server preferences.
- (vi) Vnode/VM management.
- (vii) Replication enhancements.

(b) DCE 1.2.2:
- (i) Plaintext password.
- (ii) Netware coexistence.
- (iii) Kerberos V5 support.
- (iv) Public key support.
- (v) User-to-user authentication.
- (vi) Multi-cell groups.
- (vii) Single-threaded RPC.
- (viii) Improvements to CDS.
- (ix) Improvements to security.
- (x) Multi-home support.

In addition there will be an internal release for use within the project. It will contain the DCE 1.2.1 core changes and will be used as the platform for the other development work.

4.2. Changes from Previous Version of this Document

Based on feedback from the OSF membership, better support for multi-homed servers was substituted for converting administrative lists to ACL's.

Careful readers will note that a number of items in the previous version of this document are no longer specified as deliverables. Among those items are:

(a) Uncommitted for DCE 1.2:
- (i) Active user monitoring.
- (ii) DCE management object classes.
- (iii) Definition of DCE small client.
- (iv) Event system.
- (v) Policies and procedures for cell and namespace management.
- (vi) SNMP MIB and agent.
- (vii) Support for multiple security mechanisms and algorithms.
- (viii) Synchronization between DCE and Netware user registries.
- (ix) X/Open Federated Naming.

There are two reasons for these changes:

(a) The initial contents were defined by the TPC and user requirements. As the project progressed from independent technical evaluations to more detailed schedule planning, it became obvious that not everything could be accomplished within the required budget and time constraints.

(b) IBM and Transarc sponsorships changed to a single sponsorship. As a result of IBM's acquisition of Transarc, the two companies re-examined their positions with respect to separately sponsoring the DCE 1.2 PST. The outcome of this examination was the decision that IBM would sponsor the PST with Transarc providing the technical portion of IBM's contribution in the form of the DFS features. Other earlier technical commitments from IBM have been removed from the committed contents list, though some (management features in particular) are being examined for possible inclusion in this or other DCE PST's at a later date.

REFERENCES

Note that OSF RFC's prior to number 62.0 were designated ''DCE RFC's''.

[Kohl] J. Kohl, C. Neuman, ''The Kerberos Network Authentication Service (V5)'', IETF Network Working Group Request for Comments 1510, September 1993.

[RFC 8] B. Blakley, ''DCE SIG Security Requirements'', DCE-RFC 8.1, Date TBD.

[RFC 26] J. Pato, ''Using Pre-Authentication to Avoid Password-Guessing Attacks'', DCE-RFC 26.0, June 1993.

[RFC 31] M. Karuzis, ''Supporting Threadless DCE Clients'', DCE-RFC 31.0, December 1992.

[RFC 48] R. Annicchiarico, J. Harrow, R. Viveney, ''DCE IDL with C++ Support — Functional Overview'', DCE-RFC 48.2, October 1994.

[RFC 62] K. Flowers, ''The PST Process: A Practical Guide'', OSF-RFC 62.0, Date TBD.

[RFC 68] A. Anderson, J. Wray, ''Public-Key Login for DCE 1.2 — Functional Specification'', OSF-RFC 68.1, February 1995.

[RFC 69] R. Zee, ''DCE 1.2 CDS Functional Specification'', OSF-RFC 69.0, November 1994.

[RFC 73] D. Varotsis, ''DFS Token Manager Redesign'', OSF-RFC 73.0, Date TBD.

[RFC 74] S. Berman, ''DFS Server Preferences'', OSF-RFC 74.0, May 1995.

[RFC 75] B. Lewis, ''Episode VM Integration'', OSF-RFC 75.0, Date TBD.

[RFC 76] J. Prem, D. Clevenger, ''Multi-Threading the Replication Server'', OSF-RFC 76.0, May 1995.

[RFC 78] T. Anderson, ''Episode Vnode Synchronization'', OSF-RFC 78.0, Date TBD.

[RFC 80] J. Wray, ''DCE 1.2 Certification API — Functional Specification'', OSF-RFC 80.0, February 1995.

[RFC Index] W. Tuvell, ''Index to OSF-RFC's'', OSF-RFC Index, *Latest version*.

AUTHOR'S ADDRESS

Rich Salz
Open Software Foundation
11 Cambridge Center
Cambridge, MA 02142
USA

Internet email: rsalz@osf.org
Telephone: +1-617-621-7253