ClearPath MCP: Unsurpassed Security

ClearPath MCP Release 18.0

White Paper
Table of Contents

Abstract 4
Importance of Security 5
History of the MCP 5
The Power of the MCP Environment 6
Authentication 7
Security Policy 7
Access Control 8
Audit and Assessment 8
Encryption/Cryptography 9
Program Environment 10
Summary 11
References 11
Abstract

Data and computer security is one of the major concerns of every business executive today. Every day, articles describe some aspect or impact of computer security, from data privacy to the business fallout from consumer data being lost in transit from company to repository, from cyberhacking to “zero days.” Most security analysis focuses on the products which are added to an insecure architecture to improve on its security, rather than starting with a secure architecture and building up. Choosing a platform and an architecture that has been designed from the ground up to be secure is more important than ever. The ClearPath Forward™ platform lines, both those based on the ClearPath® MCP operating system and its sister platforms based on the ClearPath® OS 2200 operating system, offer a secure operating environment for business-critical environments.

This paper will describe the ClearPath MCP architecture and operating environment and how it has been designed from the beginning to be secure and how that security has been engrained. ClearPath MCP systems also have the safeguards to protect the business critical data that numerous clients have placed on these platforms. These systems contain detailed auditing mechanisms, robust authentication options, authorization controls, system and user policy management, run time memory access controls, and easy configurability as well as other key features.

This paper details feature content which has been released up to and including the MCP 18.0 release and shows how the system software functionality has enhanced the underlying secure system architecture.
Importance of Security

Every business relies on its electronic data to survive. This data includes their customer data as well as their intellectual property, business processes, and company critical information. It allows the business to be up and running every day of the year, 24 hours a day. If a company’s computer systems are down or unavailable, then so is the business.

But in addition to the security of the data and the availability of the servers which house the data, the reputation of the company is also crucial with regard to security. Security breaches, attacks, loss of intellectual property, and negative publicity in the press greatly impact confidence in the security of the company with regard to its crucial data. The time and effort also contribute to the cost of security and your business. Being secure and staying within budgetary constraints is essential in today’s enterprises.

These computer systems must communicate with other systems and devices in order to run the business – even provide Internet-facing platforms. Security issues are introduced because the system must now decide whether or not to grant access to incoming requests and outgoing information. Having the computer systems which can handle this business-critical business is essential in today’s business environment.

The choice of platform is more important than ever in providing for a secure, feature-rich environment to implement and run your business.

History of the MCP

The MCP environment has evolved over the past few decades. It was revolutionary when it was architected, with integrated security at its core – in the system architecture and in the operating system. Since that point, it has evolved to keep up with the requirements for a state-of-the-art computer system (such as needs for larger memory, additional processors, databases, the addition of networking, support of cryptography and key management, workload management, virtualization, and the cloud).

From its beginning, the MCP operating system was designed to be a secure, robust, multi-processor, multi-user operating system, able to run many workloads simultaneously and securely. Since it was designed to be a large-scale environment, the concepts of jobs, tasks, and stack-based processes were integral to its operation. These core concepts of the “mainframe” made it ideal to run the workflows of many large companies. Building upon the security at its core, the MCP operating system enhances this core with additional security which makes it ideal to run complex business processes and tasks.

The MCP operating system is unique in many regards – it is a stack-based architecture, unlike many processor-based solutions. There is no assembly language or assembler – the environment was designed and written for high-level languages. The MCP operating system pioneered computing features that are commonplace now – virtual memory, shared libraries, symmetric multiprocessing, and a high-level job control language.

Fine-grained inter-process communication is defined between modules – fixed locations at which programs may share information. The concepts of shared system libraries were evident on the MCP environment before they were introduced in commodity servers. Entire message control systems (MCSs) could build upon these concepts to produce robust packages. Fine-grained access control (more than the traditional POSIX model) has always been available to restrict and allow availability to information. In addition, files on the MCP system have an associated type (not just the suffix of the file name), which in most cases, specifies the internal layout of the file. Changing the type on a file is not usually allowed, especially with executables or system libraries. Executables can only be made by a compiler; other files cannot be executed.

In all, the MCP environment was designed to provide security in-depth, from its core out to the user. The tenets of “default deny” and “least privilege” are prevalent throughout the design of the MCP operating system. For example, the TCP/IP firewall default mechanism is to deny all network traffic.

The concept of an “integrated stack” existed on the ClearPath MCP operating environments before it became an industry term. All system software (operating system, networking, database, compilers, et al) comes from a single supplier and is architected, designed, developed and tested together so that it works as a single image. This also extends to the auditing and logging subsystem, which is
MCP-centric and which only allows the MCP to write entries (privileged programs can ask the MCP to add entries on their behalf). The log file (called the “sumlog” in MCP terms) is a special file which cannot be modified by anyone other than the MCP.

In contrast, commodity servers started from single user models or single group or company models (trusted) and have tried to expand out to the enterprise level. These systems rely on the underlying hardware for most of the trust of the system and allow liberal interactions between programs. Commodity systems initially emphasized ease of use and open access over security and accountability. These systems were the other extreme from an integrated stack – you could use multiple products simultaneously, whether or not they were supported or tested together. The auditing and logging of multiple products into the same environment was just that – many different types of logs and security profiles – some in plain text files. As these systems try to address security, they have a large base of code to adapt to a new paradigm – one of security, which the ClearPath MCP environment has had since its inception.

Virtualization has changed the landscape. Add in a hypervisor, usually more concerned about datacenter consolidation than security, and the problem becomes worse. In commodity servers, there are now multiple copies of operating systems which can infect each other directly, or be infected through the hypervisor layer.

With the introduction of Software Series, this level of protection is even more important since there is no hardware characteristics that can be used to protect the data. It is the MCP architecture that has lasted the test of time for being a secure, state-of-the-art operating system.

The Power of the MCP Environment

The MCP operating system environment uses software-based controls to implement a trusted computing environment. This is designed to protect executing code from threats. This approach, which is different than most commodity servers, provides a higher level of security due to hardware independence. Since it is not dependent on any particular hardware feature, the software environment provides a more robust environment rather than relying solely on the underlying hardware protection mechanisms.

The MCP operating system has security integrated into the heart of the system architecture: designed into the file system, the memory subsystem, and the process structure. The MCP has several features which illustrate its premium security environment:

- Each 48-bit memory word has an associated 4-bit tag value. This tag value is hidden and protected by the MCP operating system, and describes the contents of the memory (for example, whether it contains code, data variables, or stack). This prohibits the execution of data or stack segments (a common attack vector used by viruses after they have corrupted a data buffer or program stack). Information read in from the network is placed in data buffers, which can never be executed.

- All accesses to data segments (e.g. arrays) are automatically bounds-checked. Data is usually kept in specified memory structures and not a common data segment. This feature provides that requests which are malformed (intentionally or not) do not cause memory corruption. This feature, along with the tagged memory structure, renders the buffer overflow attack vector useless on ClearPath MCP systems.

- Running programs and libraries are implemented as stacks, in which the complete operational context is kept. If a problem occurs, a stack dump occurs, which contains the entire operational context needed for investigation.

- Through coordination of the MCP hardware/firmware, MCP operating system, and MCP compilers, the return pointers on the program stack are protected against corruption by wayward program code. This is designed to prevent the code from corrupting the return pointer and injecting code into the system. The sophisticated stack architecture of the MCP environment also allows each data variable to be assigned a place in the stack at compile time, and therefore, promotes sharing of data at a fine level of detail.

- No assembly language. Assembly language gives complete control of the system to the running code. This low-level access is not needed for applications and opens the door to security violations. The MCP environment forces programming to be done with a high level language. It provides native MCP languages for some restricted access, but never gives unlimited access.
• The ClearPath MCP environment has a strong compiler-based code security model. Only code which has been generated by a system compiler can be run on the system.

• Each type of file on the system has an associated FILEKIND, which determines the use it has on the system. Only executable files can be run, and most other FILEKINDs impose restrictions on use of the information. FILEKINDs, to some extent, also denote the internal structure of the file. Some FILEKINDs are restricted (e.g. compilers, object code) and can only be created by trusted applications. The LOG FILEKIND is used for the log file (sumlog); such a file can only be created and written to by the MCP.

These features demonstrate the high-security environment of the ClearPath MCP platform.

In contrast, the commodity server environment has very few of these concepts. The system and the underlying operating systems are vulnerable to exploit, because the core operating system and system architecture wasn’t designed with security in mind. Even when patched (a process which occurs frequently), vulnerabilities may exist for years (or since the beginning – a “zero day” vulnerability). The frequent patching of a system can also cause other security vulnerabilities and increases to the total cost of ownership of the computer system. This is because the system architecture wasn’t designed with security from the start.

**Authentication**

Authentication is the validation of the identity of a person or process attempting access to a resource. This may occur as a result of an access of the computer by the network, or a local request.

The ClearPath MCP environment supports a wide range of authentication mechanisms, ranging from the simple usercode and password model to Kerberos to smart cards. All access points into the system are secured by one or more authentication mechanisms, so that there is no avenue into the MCP environment that a user can use for unauthenticated access. A custom system library (SECURITYSUPPORT) can be installed to add tailored authentication.

The ClearPath MCP environment enhances these interfaces by allowing aging of passwords for authentication and access control, requiring end users to actively protect their access into the system and supporting compliance with the Payment Card Industry Security Standard and other industry best practices. New users can be forced to change their administrator-assigned password. The MCP environment’s SUSPENDDELAY allows user accounts that have been locked out because of login failures can be automatically unlocked after a delay, with the lockout time determined by the security administrator.

The Web Transaction Server for ClearPath MCP has implemented additional mechanisms and algorithms on top of the platform functionality, to thwart hacker access and track their attempts. Attempts to access the system which are denied are counted, and that IP address and/or station will be denied from accessing the system (a system-wide option configures how many attempts are allowed). Additional options are available for delaying responses to frustrate attackers or their programs in addition to a time-based unlock feature.

The external identity that is used for authentication is kept throughout the program session – so that the audit information can trace back to the external party that was authenticated. Each audit entry contains information to allow the local identity and the original external identity to be determined.

**Security Policy**

A well-defined security policy is essential to the security of a system. Defining how the system should be configured, accessed, audited, and monitored is crucial to understanding how it works, and explaining this process to auditors. The ClearPath MCP environment has established “levels” of security inbuilt in the operating system, which specifies security requirements on the system as a whole. Four different security “classes” have been defined as steps up the security ladder from tight to the tightest security environments. These classes contain the MCP’s security options and provide settings for enabling even higher levels of security on the ClearPath MCP environment.
Security Class U, the default security class, implements basic authentication mechanisms and provides data access control to disk files.

Security Class S0 enables the security administrator to dynamically control individual system-wide security options, ranging from extensive password management, to a finer-grained authorization.

Security Classes S1 and S2 enforce the setting of individual system-wide security options, preventing them from being reset to less restrictive values. This “raises the bar” of system security to an even higher setting than the standard security environments.

The security administrator can use these security classes to implement a security policy for the ClearPath MCP system.

The definition and application of policy in the ClearPath MCP environment is made easy through the security management products of the platform. Security policies are able to be defined, modified, and applied with a few clicks. These system policies include the security option settings and logging (auditing) settings on the ClearPath MCP system as well as role-based access control for several products which run on the ClearPath MCP system.

**Access Control**

Controlling which users, processes, and tasks have access to the business-critical information is essential to maintaining a high security environment. The traditional methods of securing access are available in the ClearPath MCP environment (e.g. POSIX “owner”, “group”, “world” model). The access control model exceeds this basic level with a fine-grained access control list structure (GUARDFILES), which allows the system administrator or owner of the file to specify which users and processes can access programs, datafiles, or databases.

Guardfiles allow the file owner to specify access from the userid, accesscode, groupcode, program/process name, or codefile title. Guardfiles can also be attached to tape volumes to restrict access, as well as databases to control what database operations (called DMVERBS) can be performed. The guardfile can be applied in either GUARDED mode or CONTROLLED mode. In GUARDED mode, access control is applied to all non-privileged users except the file owner; in CONTROLLED mode, access by the file owner is also controlled by the guardfile.

In addition, each file has attributes that specify its security type and use (SECURITYTYPE and SECURITYUSE). The security use determines the kind of access when the security type is public. The default file access control (PRIVATE) restricts file access to the creator of the file; data is automatically protected from unauthorized access.

Role based access control is available for some applications in the ClearPath MCP environment. This simplifies the management of users for applications by grouping permissions (typically attributes on each user) into roles for use with those applications.

Network access can be restricted by IP address and/or userid. This can be done at many levels, from the network firewall, to the terminal server. The TCP/IP firewall implements the security concept of “default deny” by denying all traffic that isn’t implicitly allowed – fine grained access rules can be enforced, including time, day, userid, and authorized applications. The SMB service also allows access control at the userid and/or IP address level.

The TCP/IP network stack dynamically filters off all unwanted traffic by offloading the decision of whether or not to accept a packet onto the network adapters. Dynamic Port Filtering (DPF), makes the network adapters into smart filtering engines, only passing on packets for known and listening services, as well for active, open connections. It reduces the attack surface of the ClearPath MCP environment by limiting which packets are accepted.

**Audit and Assessment**

To adhere to the numerous regulations and business processes which are required in today’s world, understanding what has happened or is happening on the computer system is essential for success. The commodity servers, built up from the single user or single department, lack an integrated auditing mechanism at their core. Each process logs what it considers appropriate into a system log for viewing at a later time. Additional software may be required for this functionality.
The ClearPath MCP environment maintains a system summary log (sumlog) that can contain all events that happen on the system. This includes object creation and deletion (files, jobs, tasks, sessions, etc.), and all file access and security violations. This is part of the core operating system, and does not require any additional software or options. Access to the system sumlog information is restricted; only system administrators can view all entries. Each log record contains result and visibility information. The result field can indicate that the record contains security related data (including security relevant actions and security violations). Analysis software can filter log records based on these fields, which allows cross-product security analysis to be performed.

MCP system software uses these RELEVANT and VIOLATION bits when logging events to the system sumlog and securitylog. For example, the TCP/IP network provider uses these bits when logging its events facilitating easier discovery and analysis of network-based attacks. All sumlog entries can be translated into syslog for input into industry-standard SIEM frameworks.

Each log record contains a major and minor type that defines the content and layout of the record. This allows analysis software to efficiently filter the log for records of a specific type (for example, system messages and security violations).

Auditors, armed with the new regulations legislating control over business processes, can use the system assessment tools which are released with the system to retrieve and understand the security data on the system. These tools allow them and the security administrators to examine the system sumlog, analyze the authentication data, document the system configuration and perform other important tasks.

With the MCP 18.0 release, all security documentation has been coalesced into one document – the Security Overview and Implementation Guide (or Security Guide). This one guide provides security administrators and auditors with an indepth description of the security of the MCP environment and all of its features.

Locum Software, Ltd. provides a suite of security analysis and auditor tools (called SafeSurvey, SecureAudit, RealTime Monitor, and Safe&Secure) which can assist MCP clients and auditors to understand the security of the MCP environment. Information on these products is available at their website (www.locumsoftware.com).

Encryption/Cryptography

Encryption is a required component of any computer system in today’s world. From securing sensitive information being transmitted over the network, to securing it at rest while resident in the computer system, encryption and cryptography in general have been added to computer systems over the last few decades as computers have grown from back-room wonders to vital parts of today’s e-commerce.

The ClearPath MCP offers a full set of encryption and cryptography algorithms for both system services and user-written applications. Application writers can take advantage of this ability to add privacy and integrity features to their data, protect against tampering and helping to ensure authorized data access. The cryptographic library has been offloaded into special environments (either a hardened appliance or the system firmware depending on system type) which are optimized, hardened, and secure. The algorithms supported by the cryptographic library as well as network security protocols are constantly being updated to the latest recommended standards. In addition, deprecated or broken algorithms are removed from the libraries so that clients are not vulnerable to attack.

The standard cryptography algorithms are available in the ClearPath MCP environment.

- Data encryption: the standard symmetric AES 3DES, and RC4 algorithms are available, using key sizes up to 256 bits; as well as the asymmetric RSA algorithm, using key sizes up to 4096 bits. CBC (cipher block chaining) is supported for block mode symmetric algorithms. GCM (Galois Counter Mode) is supported for AES, especially for tape encryption. DES is supported for backward compatibility but should not be used for new applications.

- Message Digest / Hash: The SHA-1, HMAC, and SHA-256 algorithms are supported.

- Key Exchange: Key generation and management is available using RSA public/private keypairs.

- Digital signatures: Generation and verification is provided using RSA digital signatures.

- Certificates: Generation, storage, and validation of X.509 digital certificates is also provided. Certificates signed with SHA-256, SHA-384, and SHA-512 are supported. SHA-1 certificates have been deprecated according to Internet best practices.
System software uses cryptography to provide secure solutions to today’s business problems. The MCP’s native copy program, Library Maintenance, uses the cryptography subsystem to provide for native tape and CD/DVD encryption. This allows all tapes to be encrypted as they are being created, thus safeguarding the data. Tapes are automatically decrypted when read if the correct encryption key is present. External media is encrypted using the industry standard algorithm for media encryption AES in GCM mode.

Other security protocols are available for use from both system software and user applications. The Transport Layer Security protocol (versions 1.0 and 1.2) are also used with the system to provide secure networking channels between the MCP environment and the destination (for example the Web Transaction Server for ClearPath MCP can be configured to use TLS). APIs are provided to allow user-written programs to use TLS to secure their network data. SSL v3.0 has been deprecated in the environment because of known attacks against the cryptography of the protocol. In addition, certain ciphers have been removed due to deprecation. For example, TLS ciphers using RC4 encryption and SHA-1 signed certificates are no longer supported.

Other popular network security protocols are also available on the MCP environment: SSH (for use by the SFTP protocol) and IPsec (for IPv6 networks) assist the network administrator secure data in transit. The Kerberos protocol in conjunction with GSS-API is also available for secure authentication and encryption of session traffic. Kerberos authentication can support single sign-on.

All of these protocols are implemented natively in the MCP environment and therefore inherit all of the environmental protection of the ClearPath MCP environment. This reduces the attack surface because of the engrained protection of the buffer management subsystem which defends against attacks from corrupted network packets.

Program Environment

Since it is built on a secure core, the program environment of the MCP platform is a solid base for execution. System software and user-written applications can build on this secure platform to establish security environments and operate without worry about platform vulnerability.

For example, the web server for the MCP environment, Web Transaction Server for ClearPath MCP, can be enabled on a system without the need to “lock it down” if not in use. It does not give the attacker any avenues to inject code or to utilize unconfigured or misconfigured default services to gain a foothold into the system. When a worm or virus attempts to invade the system, a worker stack is assigned to the HTTP request (normal procedure for all incoming requests), and the worm or virus attempts to buffer overflow or crash the system. Due to the highly secure platform, the web service stack validates the input, notices the overflow, and terminates the connection. At worst, the stack may terminate abnormally, but since there is no “shell” or command interpreter which is running the web service, the worm or virus cannot infiltrate the system. The web server notices the stack has terminated, and creates another one in its place. The Web Transaction server also builds on the strong authentication of the ClearPath MCP environment to validate incoming requests.

Other system products leverage this secure platform to provide secure services. CIFS-based file systems (found on Microsoft® Windows and Linux systems through the SAMBA service) are simulated on the system through Client Access Services. Even though these provide commodity services, since they are running on the ClearPath MCP environment, they are highly secure.
Summary

Security is nothing new to the ClearPath MCP operating system environment. It is one of the basic tenets of the architecture, and engrained into the platform, as well as in the mind and processes of the designers and implementers of the products. Security has been designed into all of the layers, from the core operating system and trusted computing base to the user level interfaces. Because of these factors, the ClearPath MCP environment provides a highly secure computing environment for today’s mission-critical workloads.

References

ClearPath MCP Security Supports the Payment Card Industry (PCI) Data Security Standard