EMCADS™

A Technology White Paper

For more information:

www.giritech.com

info@giritech.com

CONTENTS

INTRODUCING EMCADS™

The EMCADS Server core is a transaction engine, developed to support the parsing and processing methods necessary to support the patented dynamic datagram methods.

The server core itself implements a highly modular design to allow for the maximal flexibility in future additions to the core functionality in the server, where almost any part of the core code or functionality can be replaced with very little additional adaptive development, this even goes as far as the CRC and data encryption engines.

In most repects, the Emcads server could be viewed as a proxy server, as it performs everything on behalf of the user, there are never any direct communication between a user (or a connected client) and any server or resource reachable by network from the Emcads server, it is all parsed inside the Emcads server and resources requested on the users behalf.

The EMCADS method of data processing are so efficient, since there are rarely any redundant data or unused space due to the dynamic nature, the datagrams are not only parsed and processed at the same time, saving a lot of precious processortime, but it also saves on the number of bytes transmitted over the network.

The basic modules of this unique client/server platform are:

1. EMCADS™ Clients

The basic client have quite a modest footprint and lets users quickly and securely connect to the EMCADS Server from virtually any Internet-enabled host device. The user-interface and feature set are defined by the system administrator to match specific users.

The client can be deployed on most types of personal storage media including USB keys (e.g. G/On USB), CD-ROMs, SIM Cards, SD Memory and SmartCards. This method effectively transforms “dumb” storage devices into EMCADS™ Data Carriers (EDCs).

Often a 3rd party client (e.g. a Terminal Services RDP client or a Citrix ICA client) will also be included on the EDC to facilitate use of specific network resources.

2. EMCADS™ Server

In the EMCADS Server core lies a data parsing and processing mechanism, embedding the patented dynamic datagram parsing engine.

The core design features a completely event-driven, transaction oriented processing engine for parsing the dynamic datagrams, which provides more advantages than meets the eye:

- Each user session have its own transaction queues, allowing the server to spend more time on busy users than on idle users and thereby achieve maximum efficiency in relation to resource allocation.
- The datagrams are very dynamic, a datagram always only contains the headers needed for it’s particular purpose, where the presence and/or content of any headerfield and the direction of the datagram determines the presence or absence of the next headerfield and even if the datagram (the transaction) suddenly could change from a data-transmission datagram to a signaling datagram – so more users, less overhead.
- Supports virtually any presently available encryption method and can easily be adapted for almost any encryption standard in the future.
- The validation of access occurs in two different ways, first the actual device used are validated against the ruleset to determine of it is allowed, then the user are prompted to identify and validate. Since the rules for allowing device access are completely

controlled at the server by the administrator, virtually any security policy can be enforced on the ruleset to allow access only to those adhering to the profile.

BASIC OPERATING PROCESS

A basic session consists of 4 phases:

- connection
- authentication (rights and policies)
- resource access
- disconnection

Let's look at each in detail.

Connection

1. The EMCADS™ client automatically attempts to connect to the EMCADS™ server.

   The EMCADS™ server generates a random public/private 163-bit ECC keypair, signs the public key and sends the key and the signature to the client.
2. The client validates the signature on the public key against the pre-shared certificate. If the certificate fails to validate, the client will disconnect without further action.
3. If the client validates the signature on the public key, the client will in turn generate a similar public/private 163-bit ECC keypair. The client then assembles a “Client Identity and Facility Package” - a CIF package - containing (among other things) a list of supported ciphers, hashes (for CRC), bit lengths, the symmetric key for the upstream data and some information regarding the device where the client software are running, including serial numbers for Giritech USB keys.
4. This information is encrypted with the public key from the server and returned.
5. The server decrypts and validates the package (and certificate, if used). If the validation fails, the server disconnects.
6. If the validation is successful, it will validate the client-ID, USB-KEY ID or other information submitted by the client against the ruleset configured by the administrator on the server, to determine if it should reply or deny.
7. If the client cannot pass the rules defined for access, the server will forcefully close the connection to the client.
8. If the client passes, the server will proceed to select amongst the available ciphers for the session, where the server always will select the strongest available cipher. The server also generates a random symmetric key for the downstream data, encrypts it all with the public-key from the client and forwards the data.
9. The client receives this, switches to symmetric cipher and the client and server exchange “receipts” to validate that the symmetrical engines are operational.

   As default, EMCADS™ uses 256-bit AES, the server can however support virtually any symmetrical encryption scheme, and can be configured to support multiple key-lengths, multiple schemes, cipher feedback, output forwarding for the same session and even different ciphers / keys / CRC used for up- and downstream for special applications.

The content packet is a proprietary packet format, where the entire content of the packet – which can hold both commands for the server as well as actual content – is encrypted.

TCP/IP are used for the underlying data transport, but the EMCADS protocol does not rely on any information or headers from the TCP or IP protocols, so the EMCADS communication are completely transparent to any type of NAT, PAD or even additional tunneling or encryption.

When a client wants to communicate with the server, it fills in the header fields and optionally attaches the payload, submits it to the embedded EMCADS client data management engine, which in turn encrypts the entire packet with the cipher method used for that session and sends the packet to the server.

When the EMCADS™ server receives the packet, it decrypts and validates the packet to see if it were altered in transit, processes the headers and the optional payload and performs the actions. If the packet is to be forwarded on to another user or, for example, another EMCADS™ server, it will then be re-encrypted based on that particular session.

Authentication

Currently, the only two interactive authentication schemes that EMCADS supports are the built-in and Microsoft's Active Directory (AD) thru the Windows native API.

Support for other authentication schemes – e.g. RADIUS, TACACS, Cisco AAA and biometrics – can be developed on demand.

Once the connection has been made, the standard dialog box appears asking the user to enter a login name and password.

If authentication is successful, the EMCADS™ Server immediately registers the client as online in the local user directory.

The EMCADS client also facilitates additional methods for automated logon, these are primarily used when either the client itself or other Giritech clients (such as the GRDPClient or TClient) needs to establish a session, where the most widely used are logon using randomly generated one-time tokens.

Resource Usage

Once connected to the system, users can access only those network resources they are authorized to use. This means they can see the servers, files, and peripheral devices they are entitled to use according to the rights registered against his name or group in the EDMS – but everything else on the network is invisible.

This is different from an IPSec VPN, for example, which in itself provides physical network layer connectivity, where additional filtering would require an additional firewall. It is also different from an SSL VPN which provides web-based connectivity – you have to work through a web interface.

In addition to a highly granular level of rights and embedded commands, EMCADS natively supports persistent sessions.

Disconnection

The user can select the “Exit” menu or simply remove the USB key from the host PC, the client will detect the removal of the USB key and instantly closes the connection to the server and exits.

The G/On USB client only uses the PC’s Internet connection and memory – it has not loaded any software onto the hard drive or used its resources in any other way. This means there are no usable artifacts from the session on the PC that can be used to subsequently exploit the system.

PRIMARY TECHNICAL ADVANTAGES

Security

EMCADS™ provides both a very high level of security, but certainly also a high level of flexibility for users connecting remotely to all forms of network resources. This includes:

- Strong, two-factor authentication
- Nodeless connection to the network
- Strong self-modifying encryption of all data traffic
- Server only responds to authentic clients via 1 port
- Complete control over network resource usage
- Total transparency and accountability over every aspect of the system

It is theoretically possible for an attacker to craft a client, but using a crafted client will not enable them to access the system. This is because they would have to brute-force their way into the private-key part of the servers' **and** the client's temporary (per-session randomly generated) ECC keypairs to get information on both the random keys made, encryption scheme, the checksums involved and any feedback algorithms or other self-modifying mechanisms being used.

Given the level of encryption used and the fact that all encryption keys are random, per-session generated and different for up- and downstream, this simply isn't feasible.

Further, the moment the "real client" sent data to the server (happens at least once a minute), both the crafted client's session as well as the "real client" session would be instantly terminated by the server as a perceived attempt to inject data into the encrypted data stream.

Adding biometrics for 3-way authentication would raise the level of security by providing that extra level of assurance that that user physically is who they claim to be, plus biometrics are even easier to use than having to enter a login name and password.

If a client device is reported lost or stolen, it can be instantly deactivated. The system can also be configured so that anyone subsequently attempting to launch it could be advised to return it or face the consequences. Alternatively the unauthorized user can be allowed to connect and diverted to a "quarantine" area (a "honeypot") while the system tries to establish details of the key's physical location by retrieving data about the current host device.

The fact that EMCADS™ provides a virtual resource connection instead of a physical network connection further limits the degree of access and interaction available to each user. And as EMCADS™ processes each request, every aspect of every connection can be precisely monitored and registered.

Finally, we find it worth to mention that all Giritech software, EMCADS server as well as clients, are not only encrypted but also protected using a vast range of mechanisms for detecting and combating debuggers, disassemblers, tracers, code- and API hooking tools, memory dumping or patching, entry-point modification and many other tools or techniques used by hackers to get "under the skin" on software, primarily to protect the software from being compromised by trojans or vira.

Mobility

The small form factor and high level of host compatibility means maximum mobility for users. Clients stored on a USB key, for example, will work on virtually any host device with a USB port, an Internet connection and a firewall configured to allow outward-bound traffic. This feature means users don't have to carry a laptop with them when they travel.

User Transparency

The system is totally transparent to the user. All they have to do is remember their login name and password. This simplicity eliminates the complexity of many legacy security systems that so often confuses and irritates ordinary users to the extent that they find ways to avoid or circumvent their own security measures.

2-Way Real Time Communication

EMCADS™ runs fast enough to support time sensitive applications such as IP telephony (VoIP) and videoconferencing (AvoIP). This makes it ideal for use with the wide range of both portable and stationary Internet-enabled devices that are capable of supporting live communication.

Single Point of Entry

EMCADS™ provides a single point of access for employees, partners, customers, investors, media, etc. It knows the identity of every user authorized to access the network. The location of the network is not advertised because the EMCADS™ server does not broadcast. It only responds to connection requests issued by clients that meet the first level of authenticity. This helps maximize security; it minimizes operating complexity, and makes it more cost-effective than investing in multiple systems.

Centralized Control

EMCADS™ records every detail of every connection it handles. This makes logging, tracking and tracing all the data that enters, leaves or circulates a network a simple and affordable option. It allows organizations to precisely practice true accountability over what all the system’s users do. It minimizes the need for dedicated Intrusion Detection Systems (IDS) and other network monitoring applications.

Speed, Ease and Flexibility of Integration

EMCADS™ is a “core” or “deep infrastructure” technology – an enabling technology that can plug into the widest range of existing topologies, systems and applications with little or no modification. Most of the technology is based on open standards. This, together with the modular design means it is easy to customize. Both the feature set and the user interface can be easily extended and modified at very little cost. This means preparing, installing and rolling out a fully-fledged EMCADS™ solution can be done in the fraction of the time of traditional IT security solutions.

Suitable for Enterprise Deployment

The system can be quickly, easily and cost-effectively scaled to meet increased demands. You simply add more servers. The technology also supports load-sharing and fail-safe redundancy, making it suitable for mission-critical applications.

About Giritech

Giritech A/S is a privately held company registered in Denmark. The management reserves the right to enter the content of this document without prior notice.

For more information about Giritech's EMCADS™ platform, please visit: www.giritech.com. Or contact info@Giritech.com.