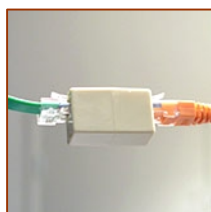
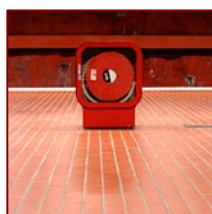


An introduction to Network security and the AppGate solution



TABLE OF CONTENTS



1	INTRODUCTION	3
1.1	BACKGROUND	3
1.2	BALANCING THE EQUATION OF SECURITY	3
1.3	SECURITY FUNCTIONS IN THE SYSTEM	4
2	THE SECURITY THREAT	6
2.1	EXTERNAL THREATS	6
2.2	THE INTERNAL THREAT	6
2.3	SYSTEM WEAKNESSES	7
2.4	HOW SYSTEMS ARE ATTACKED	8
3	PROTECTING THE ASSETS	9
3.1	TRADITIONAL PROTECTION: THE RING WALL	9
3.2	STEP 1: PARTITION THE NETWORK	10
3.3	STEP 2: ISOLATE IMPORTANT SERVERS	11
3.4	STEP 3: IMPROVE CLIENT SECURITY	13
4	THE APPGATE CONCEPT	14
4.1	REPRESENTING A SHIFT IN TECHNOLOGY	14
4.2	FEATURES AND ADVANTAGES	14
4.3	SINGLE POINT OF POWER	16
4.4	A TYPICAL SCENARIO	17
4.5	STEP-BY-STEP - A TYPICAL APPGATE SESSION	18
4.6	ESTABLISHING A SECURE TUNNEL	19
4.7	THE CLIENT SOFTWARE	19
4.8	AUTHENTICATION	20
5	THE APPGATE SERVER	22

5.1	THE APPGATE SERVER	22
5.2	MODULARITY, SCALABILITY & REDUNDANCY	22
5.3	APPGATE SERVER MODULES	23
5.4	LOGS AND ALARMS	24
6	APPLICATION EXAMPLES	25
6.1	EXAMPLE 1: SECURING LAN ACCESS	25
6.2	EXAMPLE 2: USING MULTIPLE SERVERS	26
6.3	EXAMPLE 3: APPGATE AS A PKI ENABLER	27
6.4	EXAMPLE 4: APPGATE AS A FILE REPOSITORY	28
7	SUMMARY	29
7.1	APPGATE SYSTEM FEATURES	29
7.2	APPGATE PRODUCTS OVERVIEW	30
	CONTACT INFORMATION	32



1 INTRODUCTION

The first AppGate system was delivered in 1997 to a company in the defense sector soon to be followed by a University.

1.1 Background

The first AppGate system was delivered in 1997 to a company in the defense sector soon to be followed by a University. The core of the AppGate system is still the same, made to protect information and to give controlled access.

Over the years AppGate has been able to push the limits of what is believed to be possible to deliver in one secure solution. Many times AppGate has been first to introduce features such as mobile support and integrated personal firewall, and is still the leader in many fields.

AppGate is truly unique in that sense that it provides one single solution to many security problems. Instead of having to use multiple security systems with poor integration, AppGate does it all in one box. The AppGate solution also solves many different problems for the user no matter where he/she is located or what kind of device is being used: smartphone, PDA, Pocket PC, PC, MAC or any other workstation.

A proof of the flexibility of the solution is that an AppGate system supports the smaller company needs for security as well as the global organization – with the same security solution. AppGate's smallest customer today has only 15 users, and the largest as of today has more than 30.000 users.

1.2 Balancing the Equation of Security

Networks are threatened in multiple ways, from virus attacks to unauthorised access.

At the same the time there is a demand for opening up the network for more flexible user access.

AppGate manages to balance both sides of the equation.



To be able to balance this security equation, there are four important aspects that need to be taken into account:

1. Network admission control
2. End-point security
3. Application VPN
4. Internal network security

AppGate functionality delivers all four “quadrants” of security in one easy-to-use solution. To be able to deliver the security solution demanded by customers, AppGate provides a wide range of functions that no other security company can provide in one single solution.

1.3 Security functions in the system

The AppGate system contains many features that work together to create an easy-to-use system yet a very powerful tool for the systems owner that can enforce security in a tightly defined manner. The AppGate system contains many features such as:

- | | |
|--|---|
| Authentication & Encryption | It offers strong encryption of network traffic with support for multiple simultaneous user authentication methods. The system is compatible with most third party authentication methods. |
| Roles & Rights Management | Flexible authorisation rules can specify in detail how and under what circumstances individual services should be available. |
| Application Protection | <p>The AppGate system is an application-level VPN system that supports all application protocols, not only web based applications.</p> <ul style="list-style-type: none"> - Supports multi-user systems such as terminal server solutions - The user uses the real application interface without modification - It does not have any NAT or other network traversal problems |
| Client Independence | The client software supports a wide number of client platforms, from Windows and Unix/Linux workstations and servers to telephones and PDAs. |
| Automatic client updates | When a newer version of the client software becomes available on the AppGate server, all clients can automatically be updated with the new version. This makes deployment of client software extremely easy. |

Client Check	The Client Check feature enables the security system to check the client's configuration before granting access to selected services.
Secure Single Sign On	Today it is common for users to have to log in several times in order to access an application. With the AppGate system, in many cases it is possible to avoid additional logins resulting in easier access of information.
Secure Print	Possibility for remote applications to securely print to locally attached printers.
Mobile Roaming	Roaming functionality in client offers the possibility to automatic reconnect to the server if the network connection goes down. It can be done without any user interaction with the system and can even handle a change of the IP addresses. Roaming is, of course, transparent to applications.
Secure Instant Messaging	Secure Instant Messaging for authorised users with single sign-on functionality.
Distributed Personal Firewall	A personal firewall for Windows clients with central administration through the AppGate Policy Manager. It has no GUI for the end user and is intended to be centrally managed. It can be installed together with other personal firewalls intended for user interaction, if desired.
The server acts as a Firewall	The server has built-in firewall functionality for complete protection of itself and the application server behind it.

The AppGate security solution provides organisations with a true virtual network where employees, partners, suppliers and resellers can coexist without any conflicting security or access issues. This is the starting point to create a cost-efficient, fast-moving and effective organisation where everyone can access information whenever it is needed.

2 THE SECURITY THREAT



As the need arises for more functionality and higher flexibility, security issues also become more important where functionality and mobility between the networks needs to be controlled.

2.1 External Threats

Today, most computer systems are connected to networks, and users demand access to data and remote systems more than ever before. For many organizations it is now common to publish information on the Internet, and services from internal systems need to be available to external users such as business partners, customers, mobile workers and home users. As the need arises for more functionality and higher flexibility, security issues also become more important where functionality and mobility between the networks needs to be controlled.

2.2 The Internal Threat

When corporate networks grow, most corporations eventually come to the point where the growth of the networks has made it almost impossible to control security. The solution is, in many cases, to segment or partition the network into security domains. This is a logical next step once one realizes that the management team, the economy department, different research projects, etc., should not share services and most likely not even be allowed to send network traffic to each others computers. It may seem to be a reasonably simple task to create such security domains, but in reality this task can be difficult since the logical network view seldom matches the physical network structure. For example, one project server may have team members working from different buildings, cities, countries, and even from other business partners' offices. In addition, users are mobile, team members change, new applications are introduced.

Another very common case is to be able to open up the network for access by external users. It can be own remote workers, business partners or other persons that need access to some resources located on internal systems. To be able to exactly control how and what these users should be able to access, opens up for many potential alliances and facilitates very much for internal users. A good identity management system with detailed authorization capabilities is the key to the solution.

2.3 System weaknesses

What are the main weaknesses in computers and how are they usually attacked? We can identify three major problem areas:

1. Operating systems are insecure

All operating systems contain security vulnerabilities that are publicly published.

Software products always contain bugs. This is a reality that we must learn to live with. Even small extremely well tested software modules seldom have fewer bugs than one bug per 1000 lines of code. This implies that security enforcing modules need to be as small as possible to be able to be trusted. Relying on a complex operating system containing 10 to 50 million lines of code to provide good security is, of course, impossible.

In addition, detailed vulnerability reports are published daily for system administrators. The intention is to make them aware of problems and to give them a chance to fix the problems. Obviously, these reports are also available to attackers, thus systems that are not fixed immediately can be attacked using the published information. Even if all known vulnerabilities were fixed immediately upon being reported, local configurations and administrator mistakes may remain undetected for a long time.

2. The network protocols are insecure

The major protocol, IP, was never designed for security.

Nor were its companion protocols TCP, UDP, ICMP, RIP, ARP, RARP, etc. This means that anyone who can send packets to the network relatively easily can spawn a network attack which could result in, for example, a situation where all packets sent from a file or application server were rerouted through the attacker's workstation and were then recorded and/or modified. The solution is to protect network servers from as much unauthorized traffic as possible, and to encrypt and digitally sign the network packets.

3. Applications and their protocols are insecure

Application programmers are often completely unaware of security.

It may not even be their task to be knowledgeable about operating systems and network protocols, but nearly all application writers seem to trust both the operating system and the protocols they use. In addition, they often create applications that are too complex to be fully tested; they utilize other components and modules from other developers without knowing or thinking about the security implications; they invent new protocols for client server applications that can be attacked; sending data and passwords in clear-text for example.

This opens up a wide range of possible attacks since it is usually very difficult, if not impossible, for a system administrator to have a clear picture of everything in a large system which includes all applications and their internal behaviours. In addition, an attacker will most likely insert new backdoors into the system just in case the exploited vulnerability becomes removed.

In many cases, it is only by pure luck that intruders are detected, such as when system administrators start to investigate performance problems or when

someone else, perhaps someone from another company, calls and informs that someone originating from this system is trying to attack them. The damage to reputation from a security breach can also be quite extensive when compared to the direct costs related to the intrusion.

2.4 How systems are attacked

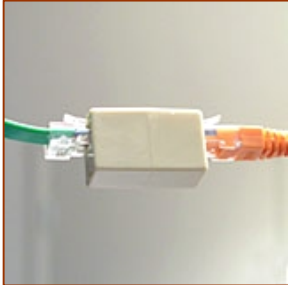
Most investigations concerning computer crime show that 60-80% of all breaches are performed by insiders. As a rule of thumb, we can say:

- 80% of all attacks come from the outside
- 80% of the successful breaches are done from the inside

An attacker normally begins by reading about vulnerability and possibly downloading some tools from the Internet, then selects a weak computer on the network and attacks it. If this computer can be breached, the attacker may find and steal a user's identity and use it to access other, more interesting systems using the breached computer. In general, intermediate computers are often used to enable new types of attacks and to obfuscate the attacks in case a mistake is made and the attack becomes disclosed. It is very easy for an attacker to reroute and record network traffic from cracked computers. Thus, these computers may be used to collect confidential information and passwords to other computers and services on the network.

The external attacker will probably not try to attack the Internet firewall since it is probably secure enough and its logs are investigated regularly. Instead, alternative paths are tried, such as telephone switches, modems or remote connections to the company. A legitimate external user who has access to a computer on a corporate network is a good target for an attack, for example a home user, a business partner, a remote sales office or even a customer. Regardless of how unimportant the system is considered to be by the system owner, for the attacker it is the opportunity to spawn new attacks against other computers on the network, now as a more privileged user! He can now begin to systematically penetrate yet other servers and other parts of the network. Using this methodology, it is possible for attackers to move from server to server and from company to company.

3 PROTECTING THE ASSETS



It is important to increase security to a reasonably high level to limit potential damages. Proper protection of systems in combination with controlled access is the solution, and there are some possible ways to achieve this.

3.1 Traditional protection: The ring wall

Perhaps the most common way to protect the corporate network and the computers it hosts is the firewall. The firewall is used to create a protective shell around the network. It assumes that everyone on the outside is evil and everyone on the inside is good, which obviously is not always the case. With this architecture, we must address the problem of finding out how to distinguish the “good people” on the outside from the not so good, i.e. find a method for granting external persons access to the inside. Traditionally this is done by opening up the protecting firewall for certain kinds of traffic, normally after some kind of authentication. But it is still not possible to offer any protection once someone is granted or has acquired access to the inside.

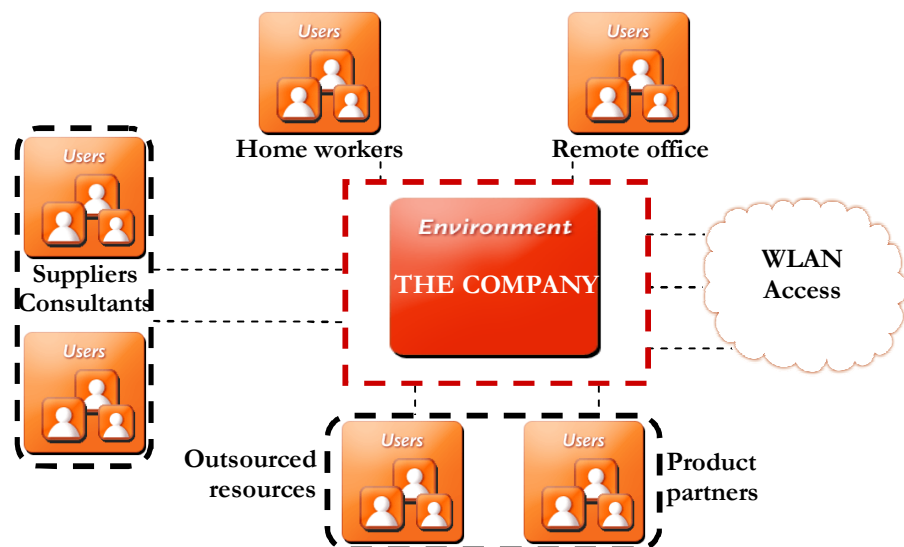


Figure 1. We can no longer hide behind a wall

This way to build systems is very similar to how cities were built in the 16th and 17th century where ring walls were built around the cities and gates with guards checked everyone passing through the gate. The same problem was present at

that time: who should be granted access and what should the credentials for access be? And again, once being admitted through the gate, the visitor was regarded as a good citizen and had the same rights as everyone else in the city. We don't build cities like this anymore. We have moved protection closer to the assets; buildings and offices have locks and access controls are distributed and is very granular. In addition, there is no difference between unauthorised persons, visitors or not.

This new way to build cities that we take for granted when protecting our own homes should also be applied to our networking environment. The firewall-centric solution was designed a long time ago and is now slowly becoming obsolete and is abandoned for more modern solutions. Protection is moved closer to the assets, i.e. closer to application servers but also closer to clients such as workstations and laptops. In the ideal world, the networks do not contain any resources or assets at all, they are just networks and transport data. Just as any street in a modern city; everyone is allowed to use it but access to resources is closely monitored. This approach also has the advantage that the "remote access" problem that used to be specially treated, is now no longer different from any other type of access. Application servers can grant access to any authorised user regardless of his/her physical location, although different kind of service may be granted based on many different parameters.

Note that the traditional firewall does not have to be removed. It can still check traffic passing by and remove obvious threats and it is a good tool for implementing defence in depth (multi-tier security). However, the final check is always done very close to each application and the correct operation of the central firewall will not be fatal to system operation. This level of protection can be seen as "border control" and is very similar to what takes place at immigration when foreigners travel to a new country. It is impossible to foresee each person's intentions but a first level of filtering is good to do and the final decision about granting access is done locally, closer to the assets. It also has the advantage of making the border firewalls relatively static. They don't have to behave differently when new services or new users are introduced to the system.

The questions to answer now are how to transform the traditional firewall-centric view into a more modern architecture and, as always, there may exist more than one solution to the problem.

3.2 Step 1: Partition the network

The first step forward is simply by observing the fact that the larger a network becomes, the more insecure it will be. Therefore, security can be improved by partitioning the corporate network, i.e. to divide it into several smaller security domains. Traffic between domains should be strictly controlled and potential problems logged. This immediately puts a limit on the maximum amount of damage a security problem can cause and increases the possibilities to both detect and deal with potential problems. It can be argued that this is not a real solution to the security problem since we still build a wall around each domain. While this is true, it can still be a good solution if the segments are small enough.

A problem with creating these security domains is to find out where the boundaries should be placed. The logical view of the network (as shown in the picture below) does not necessarily match the physical network structure where

networks span both buildings and sites and users are located almost anywhere. Anyone who has tried to implement this in an existing network knows how hard it can be. Many unexpected problems will be discovered that disturb, heavily influence or possibly even ruin the work. However, *if* this work is successful and smaller security domains can be created, the reward will be a network that is possible to maintain not only from a security perspective but also when reliability and availability is considered.

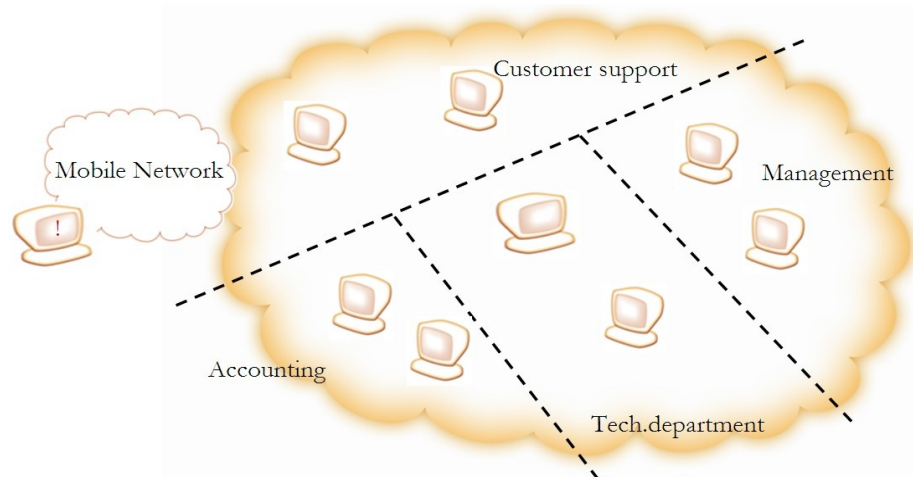


Figure 2. Security domains make it possible to handle security problems.

3.3 Step 2: Isolate important servers

The next step is to fully move away from the “ring wall” architecture, i.e. to move all application servers away from the corporate network: “what cannot be reached cannot be breached”. If the servers can be protected against all unauthorized and unwanted traffic, then operating systems, network protocols and applications cannot be attacked. This is the way the AppGate system works: AppGate servers protect one or more application servers from unauthorized traffic; they make sure only authorised users can talk to the servers; they encrypt network traffic; authenticate users and make authorization decisions and log all actions taken. In short, the AppGate system offers protection very close to the application servers and allows users to have different roles and gives access to different resources based on many different parameters such as location, time of day, authentication method and client system being used.

The system makes it possible to implement protection in the system close to the applications and can transform the networks from being assets to just become “backbones” or transport ways for network traffic.

How many AppGate servers should be used? If only one server is used as shown in the figure, there is a risk that the network behind the AppGate server becomes very large and we are back in the “firewall-centric” architecture again with one inside and one outside. If so, once granted access to one service on the inside, it may be possible to see and access other services on the supposedly secure network. This situation must clearly be addressed and there are different solutions to choose from.

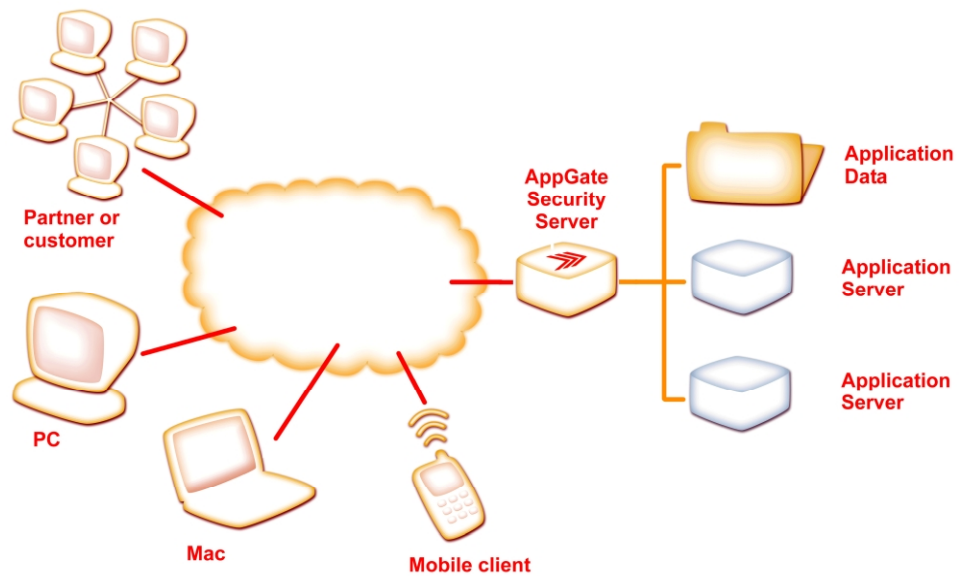


Figure 3. An AppGate Security Server is used to protect application servers from unwanted and hostile traffic and give authorized users access to resources.

Traffic on the protected network (the network behind the AppGate server) must be isolated, i.e. it should be impossible for a user on one server to reach or even see the other servers on that network without going through the AppGate server.

This situation can be solved in different ways depending on the situation:

- a) **Multiple network interfaces:** The application servers can be attached to one network port each in the AppGate server. This works if the number of servers behind the AppGate server is not too large. It offers good separation and it is impossible to reach other servers without going through the AppGate system.
- b) **VLAN technology:** Another way to implement separation is to use VLAN technology. This is especially useful in situations where switches are already present that support VLAN separation, which is typically the case in larger server parks. (This is probably the least secure solution from the ones proposed here, but may be good enough in many practical situations.)
- c) **Use of firewalls:** Firewalls such as the AppGate personal firewall can be installed on (or in front of) all application servers which are configured to drop all traffic not coming from the AppGate server. This prevents all server to server communication on the secured network. However, unless traffic is encrypted on the network, it may be possible to eavesdrop and spoof traffic on the network, although it requires hacker tools to be installed and used.
- d) **Multiple AppGate servers:** A very good solution is to rely not on just one AppGate server to protect the network but to have more AppGate servers that control different application server networks (see the figure below). Protection of each server network can then be implemented with one of the techniques above, if it is still needed. This solution is often very attractive since a distributed AppGate server system is easy to administer, load will be shared on many machines and even some redundancy is obtained since all services do not depend on just one system. The licensing system for the AppGate system does not necessarily make this solution more costly than using only one AppGate server.

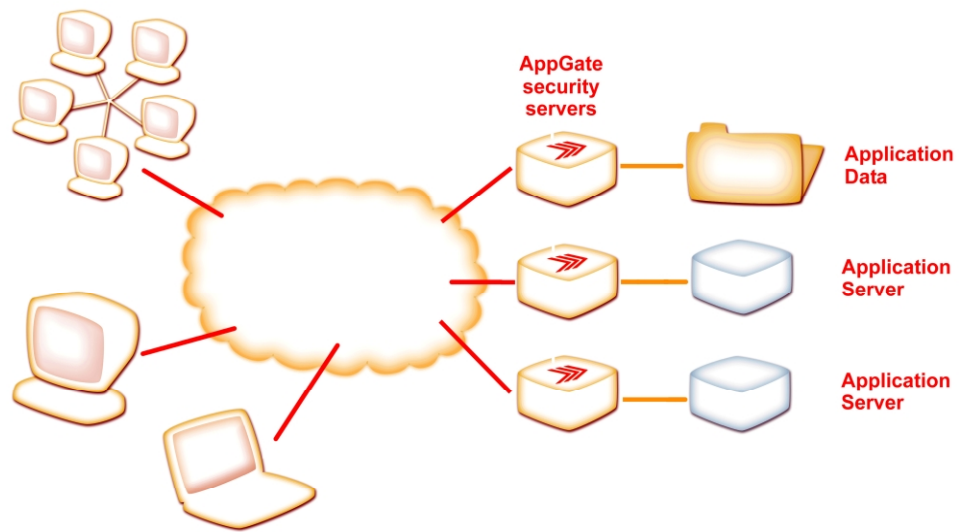


Figure 4. Three cooperating AppGate servers monitor traffic to application server networks.

The exact configuration may vary depending on security requirements, number of clients, servers and networks being used and other constraints, but the AppGate system is very flexible and is able to fit into most organisations without much configuration work.

3.4 Step 3: Improve client security



It is also important that all clients have an appropriate level of security and is able to withstand different types of threats. They need to be correctly configured, configurations must be reviewed and all software patched to make sure they do not contain any publicly known vulnerabilities. Most operating system and software vendors offer patches on their web pages.



The use of high-quality personal firewalls on workstations is becoming a necessary way to make sure malicious network traffic (e.g. IP packets with “strange” options) does not reach the operating system or applications. It helps to block unwanted traffic from the network, for example to prevent local hard disks to be accessible over the network.

The security system should also be able to do a “client check” before access to sensitive resources are granted. This check could guarantee, for example, that the client has anti-virus software installed, a good personal firewall is in use (such as the AppGate personal firewall possibly even with a specific policy in place), that no file sharing software is present or any other rules the application system owner would like to enforce before access to that application is granted.



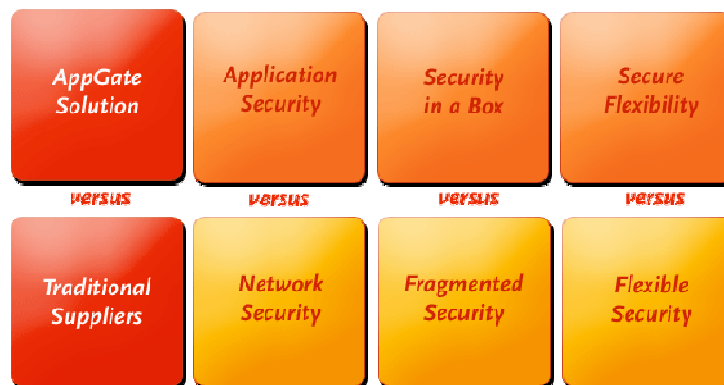
4 THE APPGATE CONCEPT

Through clustering, an AppGate solution can grow from 15 users to thousands of users without problems.

The customer benefits from a security system that's reduces complexity.

4.1 Representing a Shift in Technology

With an increasing demand for security but also for flexibility, old network based technologies quickly become a thing of the past. An example of this is the growth of application-level VPN:s. AppGate uses the advantage to build a solution that solves many security issues in one box.



The AppGate system makes it easy to protect sensitive application servers by moving them to dedicated secure networks. Accesses to applications offered from these servers will then be strictly controlled by the AppGate server. There is no way for an unauthorized person to send network packets to these application servers or even discover their existence.

4.2 Features and Advantages

The AppGate system enables system owners to simultaneously share and protect networked resources in a tightly defined manner. It combines strong encryption of network traffic with a powerful authorization system that offers system owners complete control over resources and system usage.

The AppGate system has a portal-like user which makes users unknowingly interact with the security system. When a user clicks on an icon, a message is (invisibly) sent to the AppGate server to request traffic for an application. The request is logged by the server and if it is granted, the client receives an OK and can then start the application for the user, for example a mail reader or a web browser. This mechanism ensures that only traffic to applications that are

currently needed are enabled in the security server. This also has the positive effect that help-desk and other support functions don't have to answer questions about why certain services don't seem to work.

AppGate merges different access technologies for the user. With an AppGate system it does not matter if the user is using a Mac, Windows, Linux, a Mobile phone or a PDA or if the access is done without any user intervention at all such as through applications that automatically connect to servers through an AppGate system (we have, for example, scanners that automatically connect to servers). The user can use the same authentication system regardless of the device being used. The system is flexible enough to allow users to run virtually any application, i.e. it is not necessary to change to a web interface when connecting from a foreign network. Therefore, the user is independent of the infrastructure and security is still preserved. In addition, the organization does not have to rebuild its infrastructure when a new device is added, such as a Blackberry, or have an internal change of network architecture just to support remote connections from various devices.

Since the client works with many different kind of devices, is it possible for a user to have a mobile phone that, for example, through a secure IM server receives a question from a colleague; accesses contact information from the corporate CRM system via a PDA; then sends an e-mail from the same device and then reads the answer a PC at home; then continues to work with the document graphics on a MAC inside the office or on the way getting there. Possible scenarios are unlimited.

The AppGate server offers a complete identity management system. It makes all necessary security decisions in the system, it authenticates users and authorizes them to use specified services, encrypts and decrypts network traffic, performs access control, logs events and is a central point for administration. AppGate is truly unique in that sense that it provides one single solution to many security problems. Instead of having to use multiple security systems with poor integration, AppGate does it all in one box and requires no modification of applications or application servers.

The AppGate Security Server can act as the core of the security solution. It controls user access to protected resources and its powerful and flexible authorization database contains rules for what applications and services should be available to each user. User accounts and user roles can be stored in third party servers (Radius, LDAP and AD) if desired. The server supports simultaneous use of different authentication methods for maximum flexibility.

The powerful authorization system can, for example, state that local users on the corporate LAN may access a service such as a network disk during office hours after providing password authentication, but remote users need to use a smart card for authentication and must have a personal firewall installed for the same service to be available. It is possible to state for each service exactly under what circumstances it should be available.

Remote administration of the system is possible using GUI tools with different administrator roles. Administrators can quickly add or delete users as needed, individually or in groups based on their roles, or create automated or scripted

updates when needed. All user and administrator activities are logged by the system. The logs can be very detailed if necessary, down to every byte being sent and received by an application server.

Works on many different kinds of devices. The AppGate Solution works with many different operating systems and different kinds of devices like smartphones, PDA, PC or MAC. Using CrEme - The Java™ Enabler for Windows® CE, AppGate works with many mobile devices and supports over 13 different types of CPUs and operating systems. This means that AppGate also works with handheld PCs, Pocket PC and Windows Mobile devices like Qtec, iPAQ, Pocket PC and Psion.

Alarms can be generated and be sent to external systems for immediate action through SNMP, syslog, email, pagers, etc. The AppGate server has built-in firewall functionality for complete protection of itself and the application servers behind it. It supports server clustering for scalability and redundancy, and the system scales almost linearly: one additional server gives about twice the performance, etc. Clustering makes the system ideally suited to not just give remote users access to internal resources but also for internal users on the corporate intranet.

The system is easy to implement and maintain. Application servers need not be modified and traffic can be compressed to increase performance on slower links and reduce communication costs for devices where users are charged for the amount of data being transferred.

4.3 Single Point of Power

The AppGate system can control and enforce security policies throughout the network, including the connecting devices. It merges different transmission technologies such as wireless; mobile and fixed into one secure solution. In that way the AppGate user can access vital information in a secure and controlled manner independent of whether he/she uses a PC, Mac, Mobile Phone, PDA or Linux workstation.

The AppGate system offers the system owner detailed control over resources and how and under what circumstances they can be accessed. Each resource or group of resources can have a rule set assigned. It is even possible to make checks on the client side that affects whether a particular service becomes available. The following is a possible list of parameters that can affect whether a user gets access to a particular service:

- Location of the client, such as IP address.
- Authentication method, for example a password may be enough on the internal network but a smart card or a token device is needed if the same resource is accessed over the Internet.
- Time of day and day of week, for example daytime on the local network, simpler authentication such as passwords may be accepted.
- Whether the client has a personal firewall installed. The AppGate server can even demand that the AppGate personal firewall runs a specific policy before granting access to some more sensitive applications.

4.4 A Typical Scenario

When a user wants to access a service running on a protected application server, he/she first starts the AppGate Client which sets up an encrypted tunnel between the workstation and the AppGate server. After successful authentication, the AppGate server consults its authorization database to determine which services should be available to the user. It makes the decision based on the user's identity, the current time and day of the week, the authentication method used and the client's location (IP address). For example, the authorization database may state that local users on the corporate LAN are allowed to access a particular service being authenticated with a password only, whereas remote users need to use a smart card for authentication.

Information about the currently available services is then sent to the AppGate client which presents it to the user in a window (see the figure below). The user can now select which applications or services to run by clicking an icon. When an application is selected, a request is sent to the AppGate server to enable the necessary traffic for the application to run. The server logs the request, checks the authorization database again, and if it decides to allow it, it enables the traffic. The AppGate client may now automatically start an application on the user's workstation, and if desired, the AppGate server can simultaneously start an application on the remote server. The AppGate server can also answer questions from application servers about the identities of the users connecting to them.

The client software allows users to work according to the “principle of least privilege”, which means that they will “automatically” ask for access to only the services they currently need. For example, a user who wants to read e-mail from a protected server, clicks on the e-mail icon to start the mail reader – no other traffic will be enabled until he clicks on another icon and explicitly asks for access to another service. This means that the AppGate server can and will allow traffic to only those applications which are currently in use, which in turn has some good implications:

- In the worst-case scenario – when a user’s workstation has been infiltrated – it will still not be possible for the infiltrator to access other services than the ones that the user explicitly has requested.
- The AppGate server’s detailed logging system allows system administrators to know exactly which users did access a service at a specific point in time. This is very different from just knowing which users could potentially have sent traffic to the application server.

This is important information when investigating security problems and when doing security audits since it allows the investigator to know exactly which users were using a service. In addition, when building application service provider (ASP) services, this information can be used by the billing system where customers are charged for their application usage. This applies to conventional ASPs as well as internal IT departments offering services to their users.

Another important feature with an AppGate system is that it is always the user’s identity and not the client address that determines which services are made available. Rules and conditions can be associated with every service. For example, rules may state that a particular service should only be available to “local users and on working days between 9 and 5”.

4.5 Step-by-step - a typical AppGate session

The AppGate system is intuitive for users, using these steps to initiate a user session:

1. When the user starts a session, the AppGate client sets up an encrypted tunnel to the AppGate server. The traffic is encrypted using the SSH protocol and the client software begins with verifying the servers identity.
2. The AppGate server authenticates the user, for example through a password, a certificate or a token device such as SecurID. User identities can reside on the AppGate server or on an external LDAP database server.
3. The AppGate server checks the user’s identity against a database of authorized users and access privileges.
4. The system checks what services and applications should be available to the user. Each service has a set of rules defining under what circumstances it should be available. For example, access to a network disk could be allowed after password authentication when the user is connecting from the local LAN, but may require SecurIDTM authentication when connecting from elsewhere.
5. Available services are displayed to the user in a window. By clicking an icon in the window, the service is enabled and an application can be started. For example, clicking the mailbox icon enables traffic to the mail server and launches the user’s mail program.

4.6 Establishing a Secure Tunnel

The client runs as an ordinary application on the workstation, thus it does not modify the operating system or its components. The only requirement for a client to be able to connect to an AppGate server is that it can establish a connection to the server at TCP-port 22. It does not matter what underlying networks are being used or whether the network addresses are being translated in the path by outbound firewalls doing NAT, or other such things. The client software can also traverse web and Socks proxies when connecting to a server.

When a new connection is initiated, the client begins with checking the server's identity to make sure that no one is spoofing the connection (1024-bit host keys are used complying with the Digital Signature Standard, DSS [FIPS publ. 186]). Next, the client and server agree on session crypto-keys for their connection (using Diffie-Hellman key exchange with SHA-1 as hash). Supported symmetric ciphers are 3-DES (112/168-bit keys), Blowfish, Arcfour/RC4 (both with 128-bit keys), and the new crypto standard AES/Rijndael with selectable key lengths (128, 192 and 256-bit keys).

After a connection has been set up and the user is authenticated, a secure, encrypted tunnel is created where traffic can be tunneled and multiplexed; terminal traffic, X- windows traffic, SQL requests and windows file sharing (SMB), etc. It is also possible to have data compressed before being encrypted and transmitted to increase network bandwidth. On slower links (approx below 1 Mbit/s), the performance boost due to compression can be substantial.

AppGate is based on Secure Shell, SSH, which handles traffic tunnelling and encryption. It uses an open protocol and supports strong ciphers with long encryption keys (see <http://www.ietf.org>). SSH is considered to be a mature, proven and reliable technology.



The AppGate system has been certified by the West Coast Labs and has Checkmark certifications (www.check-mark.com) in three categories: in the VPN category; in the Firewall category; and as the first certified product in the Application Gateway category! Note that it is important to use products using strong ciphers with long encryption keys, ciphers with 40- or 56-bit keys such as DES have keys which are too short to be fully trusted (see "Cracking DES" by Electric Frontier Foundation, O'Reilly, ISBN 1-56592- 520-3).

4.7 The Client Software

The AppGate client is a normal program that executes only with the user's privileges, i.e. it executes as an ordinary application as far as the operating system is concerned and it does not modify any system components such as the IP stack. The client is written in Java and makes full use of optional native code libraries (C/C++) for PKI authentication and fast encryption. The Java implementation gives extremely wide system support, such as Unix systems, Windows 95/98/ME/NT4/2000/2003, MacOS and many other platforms (for a complete list, please see the AppGate Technical Specifications).

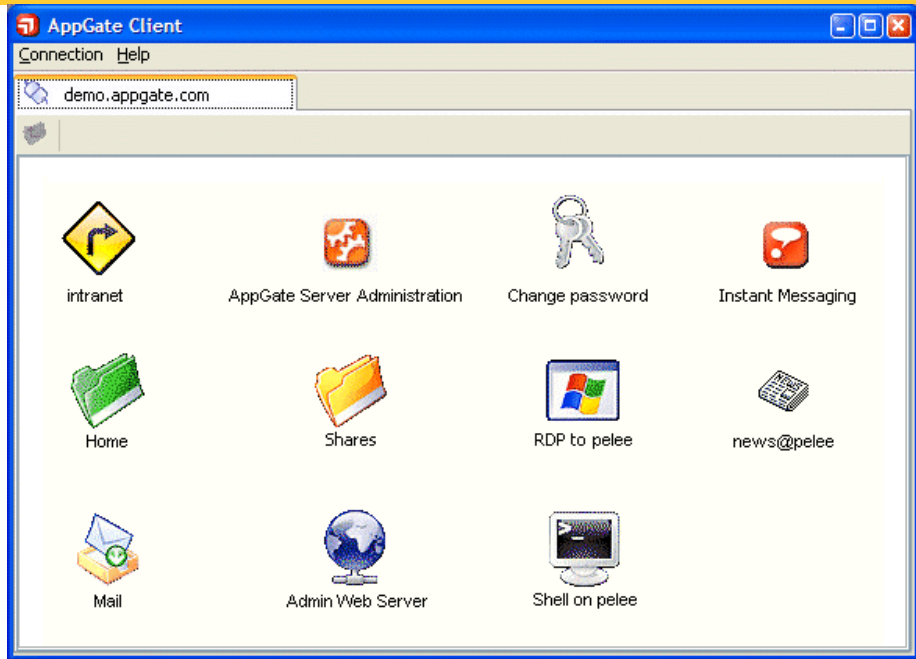


Figure 5: AppGate Client user interface with icons showing available services.

The *AppGate Client* has a full graphical user interface (GUI). Each icon represents an application or a service that is accessible to the user. By clicking an icon, a request is sent to the AppGate Server to enable traffic, and if granted, the request is logged and the AppGate Client (optionally) starts the application for the user.

The users interact “invisibly” with the security system, as they only have to click an icon to enable traffic and start an application. This way, the users of the system will only open traffic to the applications they currently are using, not to all the applications they are allowed to use. When security administrators need to investigate a problem with an application server, they can go back to the AppGate server logs and see exactly who accessed an application at a specific point in time. Service providers can also refer to these logs in order to charge users for applications they actually use, instead of the applications they have available to use.

The Applet

There is also an applet client available. It can be downloaded from a web server or the AppGate server in a normal web browser. The applet is designed to have minimal system requirements, thus it has a limited GUI and is compatible with very old versions of Java. It is especially useful in environments such as airports and Internet kiosks where there is little knowledge about what the capabilities of the systems are.

4.8 Authentication

The AppGate system supports a number of different authentication methods. Here follows a brief introduction to some authentication methods, but for a

complete list, please see the *AppGate Server Technical Specifications*. The authentication system is normally handled by a third party product, which means that some systems do not support the wide range of client platforms as the AppGate clients do. More than one authentication system may be used at the same time in one AppGate server, if desired.

Plain passwords can be used for authentication. This is the weakest type of authentication method supported since the same password is used over and over again. If a password is captured by an attacker, it is possible to impersonate the user at any time just by typing the password – there is no way for the AppGate server to distinguish this user from a legitimate user. However, passwords are simple to use and may fit perfectly in some environments. The passwords used in the AppGate environment are always sent in the encrypted tunnel to prevent them from being visible on the network.

One-time passwords are much safer since they can only be used once. Most one-time passwords are generated by some kind of gadget, for example a token card that displays a new, unique, password for each user session. One example of such a device is the SecurID™ card from RSA Security Inc., which generates and displays a new password every minute. Other similar products interface with a Radius server that contains the user authentication data.

Certificates are an increasingly popular technique based on public-key encryption as defined in the CCITT X.509 standard. A certificate is used to bind an entity's distinguished name to a public key using a digital signature. The certificate itself contains, more or less, the user's distinguished name, his/her private key and the issuer's name and public key. A certificate can either be a "soft certificate" which is a password protected file in a computer, or it can be permanently stored and protected on a smart card. Note that since soft certificates are only password protected, they are only slightly more secure than plain passwords.

5 THE APPGATE SERVER



The AppGate server never allows remote TCP packets (segments) to reach a protected network; it always unpacks the data contents, decrypts it, inspects it and then creates new TCP packets that are sent to application servers.

5.1 The AppGate Server

The AppGate server runs on Solaris (64-bit processors). Most often, the AppGate server can be treated as a black box and administration is done through the admin GUI. For advanced administrators, the system offers the possibility to easily script tasks and to easily import and export data to other systems.

All traffic through an AppGate server is carefully inspected by a filter module. Only correctly formatted packages destined for port 22 (ssh) are passed through. The operating system (Solaris), which in itself is stripped and hardened, receives the packets, unpacks them and directs the encrypted data stream to the user's decryption process. After inspection, new packages are created and sent to the application servers. Thus, an AppGate server never allows remote packages to reach a protected network; it always unpacks packet contents, decrypts it, inspects it and then creates new packages that are sent to application servers.

Most examples in this paper show an AppGate server with one incoming network interface and one outgoing. The server is, however, completely symmetric and can function with one, two, or more interfaces depending on the actual situation. A server with only one interface for example, can accept incoming encrypted traffic and then send the authorized traffic back to the same network.

Normally multiple interfaces or external switches are used to separate the traffic destined for the application servers to prevent unauthorized traffic between them. In other situations, such as when the application server is located far away from the AppGate server, it may be desirable to encrypt the traffic between them. The Solaris operating system has built-in support for IPsec encryption, which, if the application servers support it, can be used to secure this communication.

5.2 Modularity, Scalability & Redundancy

The AppGate server can easily and seamlessly handle multi-processor systems. It is therefore very easy to scale a server by adding more processors. In addition, AppGate servers can be clustered, i.e. several physically independent servers can cooperate in a cluster and be administered together. Each additional processor (whether in a multi CPU or in a clustered system) almost linearly

increases performance since the user processes are distributed evenly on all available processors (each server in a cluster is more or less unaware of the others existence). This way, everything from very simple to highly advanced fault-tolerant server solutions can be built.

An AppGate server can be divided into the following major components:

- **An encryption/decryption module** that takes care of the data received in the tunnels. Each user will have his own crypto-process, as described earlier.
- **An authorization database** containing services and rules for their use. The authorization database can be shared by many servers belonging to one cluster.
- **A log server** that logs all important events in the system. One log server can be shared by all the crypto- servers in a cluster or be distributed to be able to handle hardware failures.
- **Authentication server(s)** that contain authentication data for the users. An authentication server is normally a third party product that works together with the AppGate system.

Depending on the desired system performance, all modules can execute on separate physical servers or they can be bundled and execute on one single server.

Server hardware should be selected based on the required performance and level of redundancy. In very demanding environments, redundant hardware should be used so that hardware modules can be hot-swapped during system operation. The authorization and log servers can execute on their own dedicated servers (this is a standard feature of the software) even if it is more common to have them execute on the same server, possibly replicated, for increased availability, see the figure below. Note that for security reasons, if no log server is responding, no new services can be offered to the users.

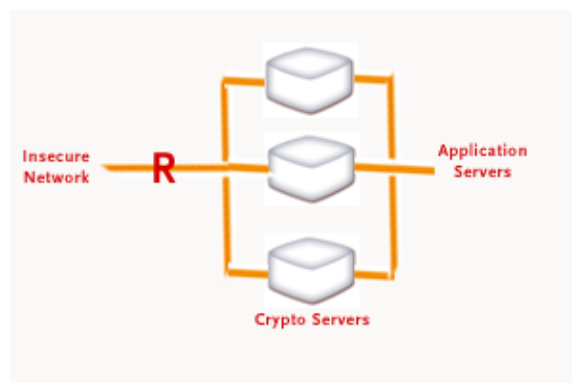


Figure 6. Example of an AppGate system with three boxes that handle crypto traffic.

5.3 AppGate server modules

To be able to deliver the highest possible level of service to the users and still enforce security, some applications or functions are specially treated by the

AppGate system. They either give additional functionality to the user or they handle a protocol which is regarded as cumbersome from a security perspective.

The IP Tunnelling Module

If an optional IP Tunnelling module is installed on a client (available in the package), it also allows applications to tunnel UDP traffic in addition to TCP. The UDP traffic is sent in the same encrypted tunnel as all other traffic. From a security perspective, UDP traffic should be used with care and not be used more than is required for applications to work.

The IP Tunnelling module allows all TCP and UDP based protocols to be used. It can even offer full connectivity between two networks, if desired.

The Secure Print Module

The secure print module is an optional module that allows applications on a remote network to print to a virtual printer on the AppGate server. The user may then authenticate him/herself on an *AppGate printer server* that retrieves the printout from the AppGate server and sends it to a local printer.

The secure print module makes sure that confidential printouts are never sent to the wrong printer and allows mobile users, such as those in remote offices, to retrieve their printed documents after successful authentication.

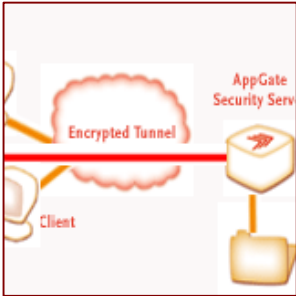
5.4 Logs and alarms

The AppGate log server receives log events from all servers in a cluster. Examples of such events are users logging in and out from the system, services or applications accessed by the users, security related events such as denied requests and protocol errors, system related information and events from the operating system (disk quota problems, hardware errors, etc.)

The log is stored as a compressed text file and can be examined with various tools.

Alarms can be generated by specifying the kind of event to look for (e.g., all failed logins or all security critical events) where an arbitrary command can be executed for each such event. This way, alarms can easily be sent through e-mail, syslog, SMS, SNMP, etc.

6 APPLICATION EXAMPLES



AppGate is truly unique in the sense that it provides one single solution to many security problems. Because of its flexibility AppGate can be used for many different occasions.

6.1 Example 1: Securing LAN access

The figure below shows users accessing protected application servers over an untrusted, insecure, network. The network can be either a corporate LAN, a WAN or the Internet. In all three cases, the communication between the users and the AppGate server is encrypted, authentication and access control is done by the AppGate server, and the application servers are protected from all kinds of hostile network traffic.

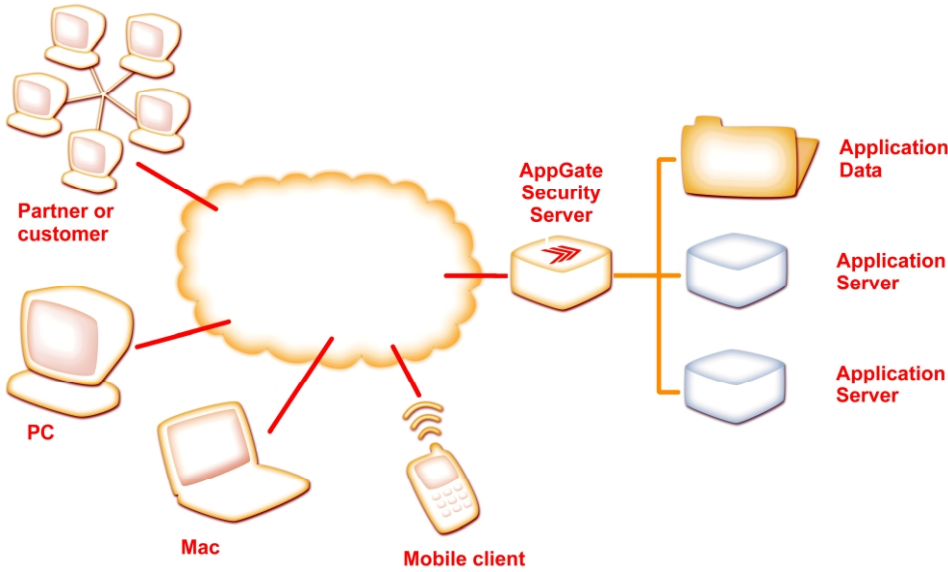


Figure 7 Using AppGate to access protected services.

The reasons behind building this system may vary, but is often the result of a security audit where a number of application servers were identified to need additional protection. Typical applications include client server applications, secure e-mail, file sharing (windows NetBIOS and Mac AppleShare), FTP (file transfer protocol), Unix X-windows applications and terminal services.

Windows-based applications can be executed remotely from protected servers with Microsoft Terminal Server (ICA) or with Citrix MetaFrame[®], both offering “virtual Windows sessions” to remote users. This makes it possible to use

workstations only to present the graphics output while the application servers are located behind AppGate servers. Running applications from remote locations in this manner means that sensitive data is never stored on the users' workstations, and in situations such as after an intrusion or theft of a workstation, there is no need to worry about what information might have been exposed or lost in the theft.

An internal or external application service provider (ASP) can use the AppGate server to give customers, users, project members, etc., access to selected applications. Due to the detailed logging, it is possible to charge the customers for their actual application usage in addition to charging for the amount of time the user is logged in.

6.2 Example 2: Using multiple servers

Instead of having one AppGate server handle all users, several smaller servers can be used to form a cluster, where each server handles traffic to one or more application servers. All machines in the cluster are administered together and they share the same log and authorization database. As far as administration is concerned, this system functions no differently than the one shown in the previous example.

In the figure below, three different project servers are used to control accesses. The users may originate from the internal network as well as from remote networks; business partners for example who need access to project data and shared applications.

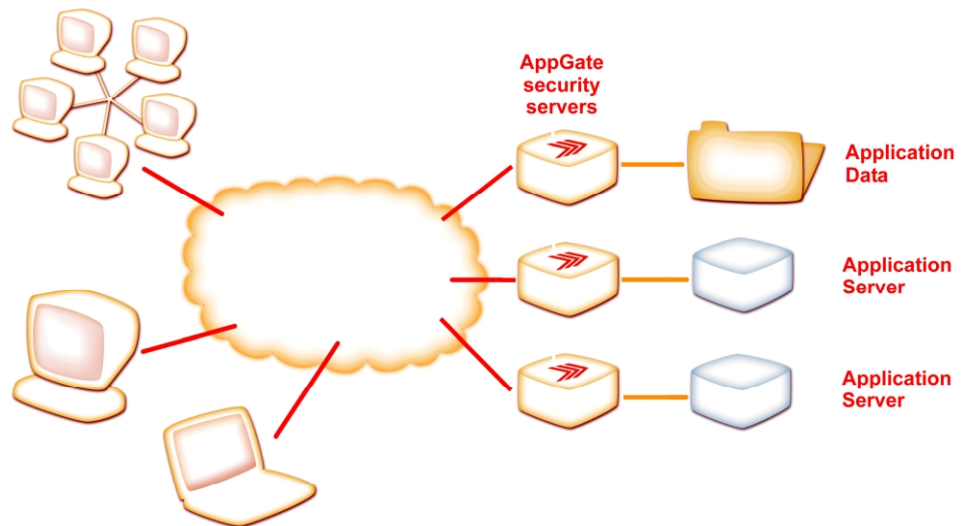


Figure 8. Three AppGate servers working in a cluster sharing log and authorisation databases.

Traditionally, it has always been a problem to give external non-employees access to servers located on internal networks because of security issues. In this case, the AppGate server will not only protect the application servers from unauthorized access, but it will also protect the corporate LAN from unwanted

traffic originating from the project servers.

6.3 Example 3: AppGate as a PKI Enabler

AppGate servers are completely symmetric when it comes to routing traffic and where users are located. All previous examples show servers with two network interfaces – one incoming and one outgoing interface. However, it is perfectly good to build systems with only one interface as shown in the figure, as well as with two, four or more interfaces. The configuration shown in the figure is applicable when application servers are located on the same network as the AppGate server. This is a typical initial installation for a customer who wants to implement a new PKI system into the organization.

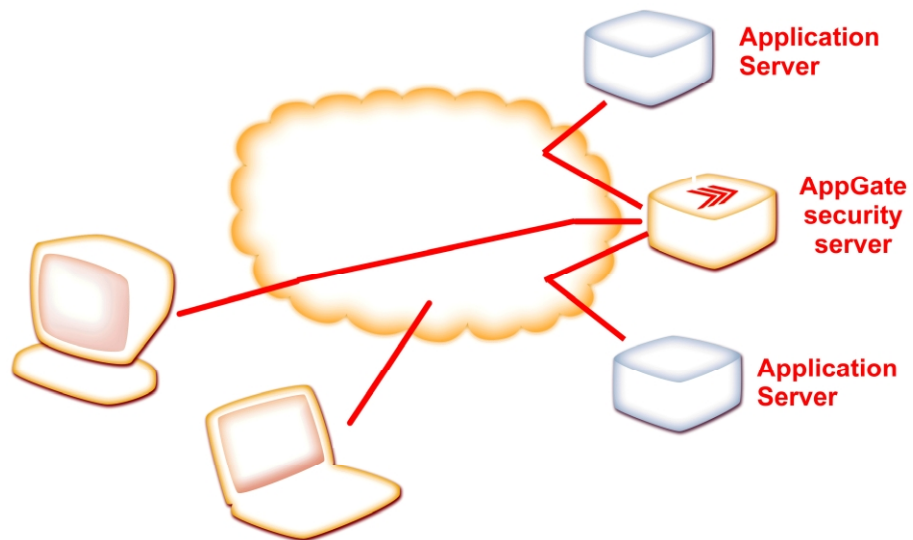


Figure 9. An AppGate server with only one interface. All traffic is sent back to application servers on the same network as the clients.

The application servers will not be aware of network encryption, access control and user authentication with the new PKI system, since this is transparently handled by the AppGate server. Users can be gradually moved into using the new authentication system; using smart cards for example. During the transition time, some users connect directly to the application server whereas more and more users connect through the AppGate server. When all users have migrated to use the PKI system through the AppGate server, the application server should be moved behind the AppGate server to be protected from malicious network traffic.

The PKI system is a third party authentication product to the AppGate system. Whether smart cards are being used or the certificates are stored in files on the clients, does not matter to the AppGate system. For a list of supported PKI vendors, please see the AppGate Technical Specifications.

If the application server and the AppGate server are not close to each other, it is possible to encrypt the traffic between them using the IPsec protocol.

6.4 Example 4: AppGate as a file repository

The AppGate server can be used as a repository that allows file transfers to and from remote users. Standard FTP clients can be used to upload and download files to the server. The FTP server can either execute on the AppGate server (a hardened FTP server is included in the software distribution) or execute on another computer on the remote network. In either case, the protocol will be examined and monitored by the AppGate server to prevent data driven attacks to the server.

The repository can either be placed on a corporate network or on the Internet, so as to allow remote users to exchange documents for example. It can also be placed between two networks belonging to different security domains to enable controlled file transfers between them. Rules can be defined in the AppGate server to control which users are allowed to upload and/or download data. All uploads and downloads are encrypted and logged in detail.

7 SUMMARY

There are four important aspects that need to be taken into account if you want to be secure; Network Admission Control, Application VPN, End-Point Security and Internal Network Security. AppGate delivers these four quadrants of security.

7.1 AppGate System Features

AppGate offers great flexibility combined with very high security:

- Access control in the AppGate server is based on individual users, not on the IP address of the client computer, although the address may affect its decision about available services.
- TCP- and UDP-based traffic can be handled when the IP Tunnelling module is installed.
- It is not necessary to modify application servers since all essential functions, such as access control and encryption, are handled by the AppGate server before the traffic reaches the server.
- Strong ciphers together with long key lengths are used for high security
- Because of its modularity, high encryption speeds can be reached. It is realistic to handle many tens of thousands of users simultaneously. It is very easy to reach LAN speed for encryption.
- Both the AppGate server and the client can start applications for the user.
- The server's functionality and performance can easily be expanded with the addition of new modules and new hardware. Highly scalable and redundant systems are easy to build.

Network Admission Control:

Rights Management
Client Check
Distributed
Personal Firewall



Balancing the Equation

End-Point Security:

Personal Firewall
Cache Cleaning
Client Check



Application VPN:

Authentication & Encryption
Roles & Rights Management
Client Independence
Full Application Support
Secure Print
Mobile VPN Roaming

Internal Network Security:

Authentication & Encryption
Roles & Rights Management
Single Sign On
Full Application Support
The server acts as a Firewall
Secure Instant Messaging

7.2 AppGate Products Overview

AppGate Security Server

The AppGate Security Server unifies all the necessary security elements such as Application Level VPN & Clientless Access, Rights Control Management, Application Protection, Client Check, Secure Single Sign On, Secure Print, Secure Instant Messaging and Mobile VPN Roaming into one easy-to-manage comprehensive solution.

AppGate clients work at the application level and are compatible with almost every conceivable information processing environment. It works with almost any application as well as with existing security solutions and can co-exist with existing firewalls, authentication systems and hardware.

AppGate Console makes the system easy to manage. Authorised administrators can quickly add or delete users as needed, individually in groups of users (roles) that share the same access permissions, or in massive updates to accommodate mergers and other organisational shifts. All user and administrator activity is logged and reported.

AppGate Mobile Security Server

Mobile Security Server is an easy, secure and cost-efficient way of making information accessible via mobile devices, without the need of rebuilding infrastructure.

The AppGate Mobile Security Server enables users of mobile devices to securely access data resources inside of an enterprise's trusted, internal network. Since mobile devices are inherently insecure, it is necessary to have a good security system in place to be able to offer internal services to users.



The AppGate Mobile Security Server is a pre-configured appliance server that is easy to install and configure. It can be installed as a stand-alone unit or in conjunction with an existing VPN system. The AppGate Mobile server can be used as a complimentary product if the existing solution does not support mobile phones or PDAs.

AppGate Distributed Personal Firewall

With the Distributed Personal Firewall it is easy to manage and build a distributed protection system that fits both small and large enterprises. The Distributed Personal Firewall is designed for both Windows workstations and servers and consists of two components, the Personal Firewall and the Policy Manager.



The Personal Firewall is designed for remote administration and has no GUI for end users.



The Policy Manager allows system administrators to define and distribute global policies for all personal firewalls in a network.

The AppGate personal firewall is designed without a graphical user interface on the client machine (user's workstation or network server). It is normally remotely configured by system administrators through the Policy Manager instead of letting local users be firewall administrators that have to make decisions about traffic filtering. Administration is normally done from one or more Policy Managers, although local administration is possible by local system administrator on standalone systems.

The AppGate Distributed Personal Firewall system is ideal to use on public systems and systems used by many users, in schools and large organisations, on internal and external corporate workstations as well as on application servers.

AppGate Secure IM



Instant Messaging is growing in popularity all over the world. The ability for instant communication and the benefit of knowing who is online has added Instant Messaging as a tool together with ordinary phone and e-mail communication.

AppGate Secure Instant Messaging works in close cooperation with the AppGate Security Server. User identity is controlled in the AppGate Security Server and almost all authentication methods on the market today are supported to guarantee user identity. The Secure IM system will receive the users' IDs from the AppGate server (single sign-on) which makes it extremely easy for the users to use. It also prevents users from using aliases and faked identities in the system.

All communication over the network is encrypted. This is especially important if the system should be used both by local and remote users. Access to the IM system can be granted to all or to selected users in the account database.

AppGate Secure Instant Messaging is easy to implement. The AppGate client software is automatically downloaded when the users connect to the system. The client software supports most types of systems, but open standards ensure that also third party clients will work with the system.



CONTACT INFORMATION

AppGate headquarters is located in Stockholm, Sweden, but also has offices in other countries like UK and US.

AppGate Network Security

sales@appgate.com

www.appgate.com

Sweden • Phone: +46 31 77 44 350

US • Phone +1 919 469-5066

UK • Phone +44 (0)870 4460 223