Weak Links Mar Secure Messaging System

Scott Welch

By Scott H.E. Welch

Why IT Departments Fail On Security

With the heightened focus on security, building a secure messaging system is more important than ever. Paradoxically it is also tougher than ever. In addition, the cost of building such a system can be an expensive process for most organizations as they may have to purchase additional hardware and software tools and assign personnel to implement them.

However, even with the vast number of security tools at our disposal today, the best tools can be rendered useless unless we take a holistic approach to security. IT departments have spent the past few years looking at details such as the number of bits of encryption or the list of acceptable passwords, while ignoring glaring holes in security that are easily exploitable by even the most technically-challenged. Simply put, instead of focusing on individual components it is far more critical to focus on the end-to-end security of the whole system.

With this in mind, how can we solve these potential security risks? By looking for a messaging system that has been designed for security from the outset.

Most of you are probably running Microsoft Exchange as a messaging server. A high percentage of your end users are using laptops, with Outlook installed on them. When your users are traveling, they either use web access to your Exchange server, or they auto-forward to a public web-accessible messaging service such as Hotmail.

Does this sound like your organization? Perhaps we should look into what's really happening, from a security point of view.

Firstly, let's look at the weakest link – the end user's computer. Why? These machines are the most difficult for the IT department to control, manage, and backup, since they are scattered around the organization. And yet, this is where we find Outlook, which proceeds to place all of the user's new mail, filed mail, account information, passwords, and address book entries on the hard disk of their computer!

Basically, organizations are at the mercy of their employee's hard drive as many build a knowledge base and simply store it on the hard drive of their laptop with reckless abandonment. Imagine for a second that your accounting package stored each order on the clerk's machine, or your ERP system saved data out on the shop floor – those ideas are so ludicrous that you would not consider them for a moment. But in the case of mail we've all just sort of agreed that we'll ignore that big pink elephant sitting in the room. It also goes without saying that since these end-user machines are not backed up, users lose and/or delete their mail with regularity, so that you need to budget about one messaging support person for about every 500 end users. Of course, every time a user gets a new computer you also have to invest hours to set it up correctly and to move their mail, address book, etc. over from their old machine.

While we are still on the subject of the end user machine, we have to consider viruses. Since all of a user's data, including their address book, is stored out on their machine, it becomes trivial to write a little script that when run will replicate itself by sending mail out to every entry in their address book. To make it even easier for virus writers, Microsoft helpfully includes not one but two excellent virus hosting engines in

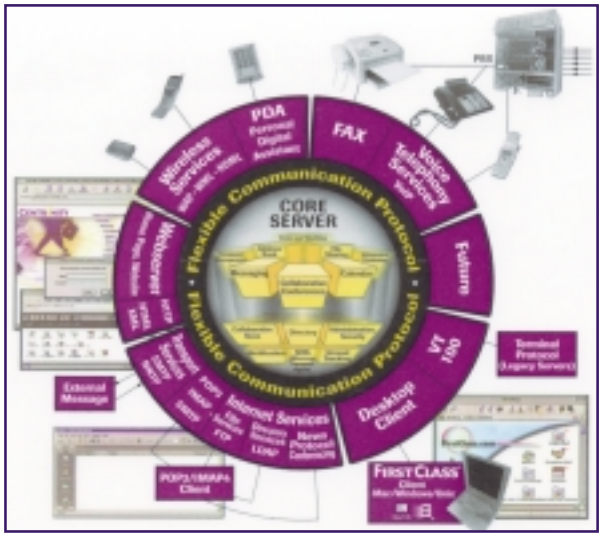

Figure 1. FirstClass™ - secure architecture

every copy of Windows: the .BAT batch file interpreter, and the .VBS Visual Basic Script interpreter. The results of this are predictable: a steady stream of viruses that send private documents out to the wide world, and at the same time massive lost productivity when people are unable to use their machines.

Now imagine no longer being held captive by a desktop or laptop. Imagine being able to securely transfer and access mission-critical data from the device of your choice – regardless of operating system, location and device. In a secure messaging system this is possible as the end-users data is always stored on the server. (See **Figure 1**.) In one simple move you now have moved the data to a secure, managed environment that you control. Plus, since the data is not stored out on the laptop, users can't mess it up. It also means that users can walk up to any machine and access their messages, filed mail, address book, etc. (sort of like the good old days, when we used terminals and a mainframe!). Add in data encryption between the client and server and you're well on your way to building a secure messaging system.

Next, since that user's data is stored on the server, not on the client machine, viruses simply lose their potency. Sure, they can run on the machine, but since the user's data is unavailable there is no way that the virus can send out data or access the address book to replicate itself. Result? Viruses die a quick death.

The next weak link is web access to mail. Web browses are ubiquitous, convenient, and completely and totally untrustworthy. When your users are traveling, they will be using browsers in airport lounges, web kiosks, customer's machines – and you have no control at all over what those browsers are or how they are configured. Lots of web kiosks have older browsers with minimal or non-existent support for encryption, and even those that support SSL have a nasty habit of caching data. Don't believe me? Walk up to a public web browser and have a look at what's in the history.

If you turn off web access, your users will immediately find a workaround, and their favorite is to simply auto-forward all of their mail to a Hotmail account. When you take a step back and think about it, your organization's knowledge base and most sensitive internal documents are being sent with abandon by the executive team over the Internet – completely unencrypted – to a system that the user has chosen because it was free. This can be a very costly mistake to any company, regardless of size.

Through a more secure remote system, users now have a legitimate need to access their content from many locations, including from their colleague's desk to their home office to half way around the world. In a truly secure system the same secure client that they use in the office would provide transparent access to the server (across an encrypted link, of course) from any machine in the world. Since the client would come from the same vendor as the server, the security guarantee would be the same for remote access as it would be for access right from the users desk. Of course, it practically goes without saying that if you want to guarantee security you should not allow web access. It is impossible to guarantee the security of remote web browsers.

In summary, as IT professionals we have to look beyond the hype to make sure that we are providing appropriate security to our organizations. We have to understand what security weaknesses we face, but we also have to keep in mind that we are providing a service to our end users. The challenge is to find a secure system that still provides the feature set that lets the end users work the way they want. TDG

About the author: ***Scott Welch*** *(scott@centrinity.com) is Chief Evangelist of Centrinity, Inc., developers of the FirstClass™ communications system. He invites your comments about security and system architecture.*