File Transfer Security
FTP: the enemy within

Open Text Connectivity Solutions Group
November 2009

Abstract

A leading industry analyst was quoted saying: “Most companies have little idea how pervasive FTP activity is in their organizations because FTP is no longer just a protocol for internal and external file integration mechanism”. Not only has FTP widely spread throughout organizations, but its popularity seems to have no limit and is continually increasing. Unfortunately there has been an increasing number of security incidents where FTP was identified as a critical factor. To put it simply, FTP is probably the least secure method an organization can use to transfer its information from one point to another.

Industry standards and government regulations such as Sarbanes-Oxley, Payment Card Industry Data Security Standard (PCI-DSS), Health Insurance Portability and Accountability Act (HIPAA), Gramm-Leach-Bliley Act (GLBA) and Federal Information Security Management Act (FISMA) require organizations to constantly strengthen the protection of mission-critical information such as credit card records, patient data or customer information. With billions of dollars of annual losses attributed to security breaches, corporations are under increasing pressure from auditors and shareholders to eliminate non-secure legacy systems.

This paper explores the business and technical reasons why companies should stop using FTP as their file transfer protocol. It demonstrates how Open Text Secure Server, the high-performance secure shell server from Open Text Connectivity, can be deployed alongside its desktop counterpart, Open Text Secure Terminal, to replace FTP, allowing companies to offer the same flexibility and convenience as traditional FTP but in a much more secure environment.

Contents

Introduction

FTP stands for File Transfer Protocol. It finds its root at the origins of the Internet when it was commissioned by DARPA in the late 1960's. In essence, FTP is a network protocol for exchanging and manipulating files over a TCP network. It follows a client-server model where an FTP client exchanges command with an FTP server to manipulate and transfer data.

FTP is a standardized protocol by the IETF (Internet Engineering Task Force) through RFC (Request for Comment) 959, replacing the original RFC 114 published in April 1971. The fact that it is an IETF standard coupled with the broad availability of many commercial and non-commercial FTP clients and servers, on a wide variety of platforms, have made FTP a very popular protocol.

A leading industry analyst was quoted saying: "Most companies have little idea how pervasive FTP activity is in their organizations because FTP is no longer just a protocol for internal and external file integration mechanism".

Not only has FTP widely spread throughout organizations, but its popularity seems to have no limit and is continually increasing. A 2008 survey from analyst firm Hilty Moore & Associates commissioned by vendor Sterling Commerce found that 64% of surveyed companies were poised to do more file transfers in 2008 than in 2007.

Unfortunately, there's nothing in IT that's black or white and FTP is certainly no exception. To put it simply, FTP is probably the least secure method an organization can use to transfer its information from one point to another.

This paper will explore the business and technical reasons why companies should stop using FTP as their file transfer protocol. It will then show how Open Text Secure Server™, the high-performance secure shell server from Open Text Connectivity Solutions, can be deployed alongside its desktop counterpart, Open Text Secure Terminal, to replace FTP, allowing companies to offer the same flexibility and convenience as traditional FTP but in a much more secure environment.

Security Breaches

340 millions compromised records and counting

Anyone remotely interested in IT security has probably noticed a surge in reports of security breaches over the last few years. This is due in part to increased pressure for transparency from the public and lawmakers but also because cybercrime has risen to such levels that it has become a major topic for news reporting.

Also, there are hundreds of places where one can get information about security breaches. One that is particularly interesting is Privacy Rights Clearinghouse. PRC is a nonprofit consumer organization with a two-part mission - consumer information and consumer advocacy. It was established in 1992 and is based in San Diego, California.

The PRC web site keeps a chronology of data breaches in the United States of America. Their listing is fairly exhaustive and although it does not contain much technical information about the nature of the breaches, it does a good job at listing how many records were compromised. In general, those records contain social security numbers, account numbers, driver’s license numbers or credit card numbers.

According to the PRC report, the total number of records containing personal information involved in security breaches in the U.S. since January 2005 amounts to a staggering 340,102,273 as of November 19, 2009. For the month of October, 2009 alone, they have documented 17 security breaches incidents involving around 76 million records. That’s an average of one security incident every two days, in the United States alone and only when the incident and the number of compromised records are known.

High-profile security breaches

In case you are still reading this document and not frantically trying to reach your Chief Security Officer to find out if your company was involved into any of these cases, there is another place that you can check out for more information: http://datalossdb.org/.

This site, successor to the former attrition.org, is a community effort under the auspice of the Open Security Foundation aimed at documenting known and reported data loss incidents world-wide.

What is interesting about datalossdb.org is that their records of security incidents are very detailed. Not only do they track the number of compromised records (when known), but they also relate each case with similar cases, identify the breach type and its source, follow-up on subsequent lawsuits, data recovered or arrests, keep a chronology of events and finally document the case with third party sources such as media news reports.

Below is a reproduction of their top 10 incidents list as of September 27, 2008.

Records	Date	Organizations
130,000,000	2009-01-20	Heartland Payment Systems
94,000,000	2007-01-17	TJX Companies Inc.
90,000,000	1984-06-01	TRW, Sears Roebuck
76,000,000	2009-10-05	National Archives and Records Administration
40,000,000	2005-06-19	CardSystems, Visa, MasterCard, American Express
30,000,000	2004-06-24	America Online
26,500,000	2006-05-22	US Department of Veterans Affair
25,000,000	2007-11-20	HM Revenue and Customs, TNT
17,000,000	2008-10-06	T-Mobile, Deutsche Telekom
16,000,000	1986-11-01	Canada Revenue Agency

FTP Cases

You might be thinking that this sort of security breach is way too high-profile to involve something as trivial as FTP. You couldn't be more wrong. Below are a few samples of security breaches where FTP was either the cause of the breach or became an easy target once the attackers gained access to the FTP credentials

New Mexico Administrative Office of the Courts

For 8 days in late May 2006, an unsecured document was exposed on the agency's FTP site on the state's computer server. It contained names, birth dates, Social Security Numbers, home addresses and other personal information of judicial branch employees. The breach affected 1500 employee records.

Texas Woman University

In late 2006, Texas Woman's University notified approximately 15,000 students that their personal data had been exposed to potential identity theft. University officials discovered that IRS 1098-T Tuition Statement data for 2005 was transmitted to an outside vendor via a non-secure connection. The personal data included names, addresses and Social Security Numbers.

Diebold Election System

Diebold Election Systems, which built the AccuVote machines used for the US presidential elections in 2002, had been parking files on an unprotected public FTP Server. Thousands of files were available: election files, hardware and software specifications, program files and voting program patches.

Though the address was obscure, people found the FTP site using a simple Google search. A Global Election Systems web site, located at http://www.geocities.com/Tokyo/Towers/2256/ (now dead) contained a list of links like "History," "Press Releases," "Staff" and -- amazingly -- "FTP."

The FTP server gave total access to anonymous users, allowing anyone to download and apparently, upload to the server. The FTP site contained no copyright statement, asked for no user name, and put no locks on directories. Visitors from anywhere in the world could simply walk in the front door.

Dreamhost

In June 2007, users of web provider Dreamhost received an e-mail informing them that someone had exploited a flaw in their administrative console software to gain access to 3,500 FTP accounts credentials. The intruder was using that information to modify the index page of the compromised accounts web site for Search Engine Optimization purpose.

SAIC

SAIC, a Pentagon contractor revealed that a file containing sensitive information about 580,000 US military personnel had been transmitted via FTP using an unsecured connection, potentially exposing the data to eavesdropping. It’s not been determined whether that data was actually compromised or not, but strict regulations forced SAIC to reveal the security issue a kind of publicity they would certainly have preferred to avoid.

FTP, a Risky Business

FTP, because of its special role in the IT infrastructure as one of the main vehicles to move information, is of significant concern as it puts a great number of business critical sensitive data at risk such as:

- Intellectual property
- Credit card numbers
- Social Security numbers
- Financial information

While Security was receiving more and more coverage from the media, it also started to make its way into a lot of other places from government office to auditors, to finally reach the core of the business world: corporate boardrooms. Corporate leaders quickly found out that lack of Security carried a heavy cost.

Cost of data leakage

The first element that speaks to any company is the cost factor. A 2007 study from the Ponemon Institute, a privacy and information management research firm from Traverse Michigan, found that the cost of data breach incidents was $197 per record, an increase from 2006 findings of $186 per record. The biggest chunk of that cost came from lost business opportunity which rose from $98 in 2006 to $128 in 2007.

Another interesting survey result was the list of the six security technologies that were enacted after a security breach. They were, by order of importance:

1. Expanded use of encryption
2. Data loss prevention solutions
3. Identity and access management solutions
4. Endpoint security controls
5. Security event management solutions
6. Perimeter controls

Interestingly enough, points 1 and 3 directly address flaws inherent to the FTP protocol.

A 2007 Forrester study on the cost of security breach found that the cost could amount between $90 and $305 per record. The gap is a result of the many hard and soft costs that can vary from incident to incident. This means that if ten records are compromised, it can cost between $900 and $3,050; 1000 records can cost between $90,000 and $305,000.

Those costs are spread among several categories:

Category	Description	Cost per record
Discovery, notification and response	Legal counsel, mail notification, calls, call center, discounted product offers	$50
Lost productivity	Employees diverted from other tasks	$20-$30
Opportunity cost	Impact on existing customers and difficulty getting new ones	$20-$100
Regulatory Fines	FTC, PCI, SOX	$0-$60
Restitutions	Money put aside in case breaches are discovered	$0-$30
Additional Security and audit requirements	Security and audit requirements levied as a result of a breach	$0-$10
Other liabilities	Credit Card replacement costs, civil penalties if fraud can be traced to the breach	$0-$25

Regulations

The next element that has garnered the attention of C-level executives is the dramatic change of regulatory landscape that has happened in the last ten years. A number of new regulations and industry standards have emerged, that have given organizations more reasons to hone their Security policies. Below is a current list of some key regulations that were introduced. Most of them apply to the United States but exist in similar form under another name in other countries.

PCI DSS

- Payment Card Industry – Data Security Standard
- **Scope**: International – Any vendor or organization handling payment card data
- **Objective**: securing processing, transmission and storage of payment card data and related information
- **Impact on FTP usage**: all payment card data and related information must be safely transmitted when in transit

GLBA

- Gramm-Leach-Bliley Act
- **Scope**: United States – Financial services companies
- **Objective**: protecting NPPI (Non Public Personal Information) in financial services
- **Impact on FTP usage**: transmission of NPPI must be done securely using encryption and authentication

SOX

- Sarbanes-Oxley Act
- **Scope**: companies whose stock is publicly traded on a US stock market
- **Similar laws in other countries**: J-Sox (Japan), Bill 198 (Canada), CLERP9 (Australia), LSF (France), L262/2005 (Italy), DCGK (Germany)
- **Objective**: enhanced corporate accountability through tight controls on financial reporting process including protection of data that participates to the companies' financial reports
- **Impact on FTP usage**: use strong authentication, access control, data encryption and data integrity mechanisms to fulfill control requirements over sensitive financial information

HIPAA

- Health Insurance Portability & Accountability Act
- **Scope**: United States – Healthcare industry
- **Objective**: protecting and privacy of patients electronic data and information
- **Impact on FTP usage**: encrypt transmission and authenticate access to patient electronic records

SB1386

- California State Bill 1386
- **Scope**: United States – California
- **Objective**: mandatory disclosure of breaches involving unencrypted personal information to any resident of the State whose data is believed to have been disclosed
- **Impact on FTP Usage**: do not allow any personal information to be transmitted unencrypted

DPA

- Data Protection Act
- **Scope**: UK – all industries and businesses
- **Objective**: protection of personal information
- **Impact on FTP usage**: securely transmit personal information with proper encryption and access authentication

Auditor and shareholder pressure

With all the publicity surrounding data breaches, it has not been long until shareholders have voiced their concerns in the boardroom and asked companies to adopt more stringent security measures to protect their data.

When it comes to FTP, auditors have been particularly vocal have kept exposing the many issues that they see with continuous use of FTP on corporate networks:

- **Increased exposure**: with many FTP servers suffering from well documented exploits on public web site, just the fact of having one of those on the company's network increases the risk and potential damages an attacker could cause.
- **Unsecured data transmission**: because it transmits all data in clear text over the network, FTP offers too easy a vehicle for eavesdropping on sensitive information
- **Lack of control over sensitive data transmission**: FTP has taken a life on its own in many corporations with servers popping up everywhere and not being managed by the IT department resulting in absence of control over the server's configuration and the information available
- **Anonymous FTP usage**: this mode, available on all FTP servers, allows a user to connect to the server without credentials, browser and download information and even occasionally upload it. This mode is turned on by default on many servers
- **Sharing of credentials**: because it only uses username and passwords to authenticate users, FTP has no mechanism to prevent multiple people from sharing the same credentials, often because it's perceived as more practical by those who do this.

FTP Weaknesses Exposed

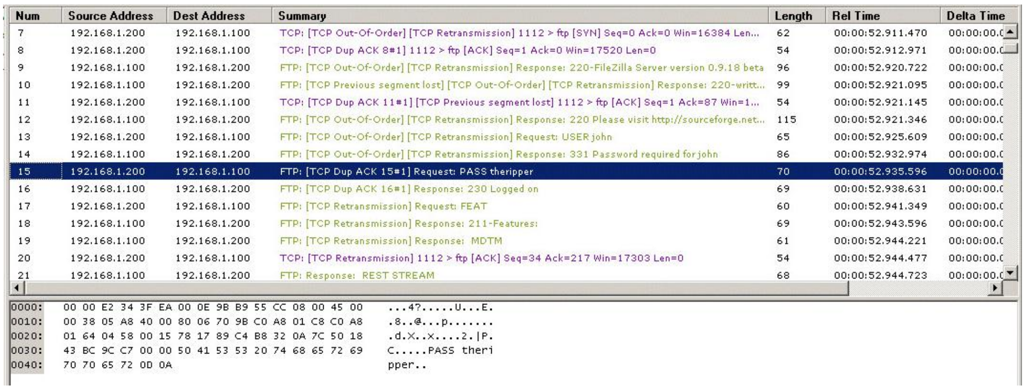
You should be convinced by now that FTP security issues are not a rarity and require immediate attention unless you want to pay the high price or get an earful from your auditors. But what exactly are the technical problems that make FTP such a poorly secured protocol? The next few paragraphs will give you an overview of the security issues of the FTP protocol.

Clear-text transmission

This is certainly the most documented and well-known problem associated with FTP. And yes, it might be news to you, but every data you transmit over FTP is transmitted in clear text, including your username and password.

You need to remember that FTP was designed at a time when network access was reserved to a few select organizations and security of what was going over the network was far from being a major issue.

Unfortunately, this fundamental issue is a major problem today, since every piece of data that's being transmitted, including credentials, can easily be eavesdropped upon. And don't be lured into believing that network sniffing is reserved to elite. Network sniffers are abundant these days and do not require any more skills to be used than it takes you to establish your FTP connection.



Num	Source Address	Dest Address	Summary	Length	Rel Time	Delta Time
7	192.168.1.200	192.168.1.100	TCP: [TCP Out-Of-Order] [TCP Retransmission] 1112 > ftp [SYN] Seq=0 Ack=0 Win=16384 Len...	62	00:00:52.911.470	00:00:00.0
8	192.168.1.200	192.168.1.100	TCP: [TCP Dup ACK 8#1] 1112 > ftp [ACK] Seq=1 Ack=0 Win=17520 Len=0	54	00:00:52.912.971	00:00:00.0
9	192.168.1.100	192.168.1.200	FTP: [TCP Out-Of-Order] [TCP Retransmission] Response: 220-FileZilla Server version 0.9.18 beta	96	00:00:52.920.722	00:00:00.0
10	192.168.1.100	192.168.1.200	FTP: [TCP Previous segment lost] [TCP Out-Of-Order] [TCP Retransmission] Response: 220-writt...	99	00:00:52.921.095	00:00:00.0
11	192.168.1.200	192.168.1.100	TCP: [TCP Dup ACK 11#1] [TCP Previous segment lost] 1112 > ftp [ACK] Seq=1 Ack=87 Win=1...	54	00:00:52.921.145	00:00:00.0
12	192.168.1.100	192.168.1.200	FTP: [TCP Out-Of-Order] [TCP Retransmission] Response: 220 Please visit http://sourceforge.net...	115	00:00:52.921.346	00:00:00.0
13	192.168.1.200	192.168.1.100	FTP: [TCP Out-Of-Order] [TCP Retransmission] Request: USER john	65	00:00:52.925.609	00:00:00.0
14	192.168.1.100	192.168.1.200	FTP: [TCP Out-Of-Order] [TCP Retransmission] Response: 331 Password required for john	86	00:00:52.932.974	00:00:00.0
15	192.168.1.200	192.168.1.100	FTP: [TCP Dup ACK 15#1] Request: PASS theripper	70	00:00:52.935.596	00:00:00.0
16	192.168.1.100	192.168.1.200	FTP: [TCP Dup ACK 16#1] Response: 230 Logged on	69	00:00:52.938.631	00:00:00.0
17	192.168.1.200	192.168.1.100	FTP: [TCP Retransmission] Request: FEAT	60	00:00:52.941.349	00:00:00.0
18	192.168.1.100	192.168.1.200	FTP: [TCP Retransmission] Response: 211-Features:	69	00:00:52.943.596	00:00:00.0
19	192.168.1.100	192.168.1.200	FTP: [TCP Retransmission] Response: MDTM	61	00:00:52.944.221	00:00:00.0
20	192.168.1.200	192.168.1.100	TCP: [TCP Retransmission] 1112 > ftp [ACK] Seq=34 Ack=217 Win=17303 Len=0	54	00:00:52.944.477	00:00:00.0
21	192.168.1.100	192.168.1.200	FTP: Response: REST STREAM	68	00:00:52.944.723	00:00:00.0

```
0000:   00 00 E2 34 3F EA 00 0E 9B B9 55 CC 08 00 45 00    ...4?.....U...E.
0010:   00 38 05 A8 40 00 80 06 70 9B C0 A8 01 C8 C0 A8    .8..@...p.......
0020:   01 64 04 58 00 15 78 17 89 C4 B8 32 0A 7C 50 18    .d.X..x....2.|P.
0030:   43 BC 9C C7 00 00 50 41 53 53 20 74 68 65 72 69    C.....PASS theri
0040:   70 70 65 72 0D 0A                                  pper..
```

Fig 1 – Capture of an FTP password using a network sniffer. The password appears in clear text in the lower pane after the FTP command PASS

Weak authentication

The second most common reproach made to FTP is its lack of strong authentication. In addition to offering an often-by-default anonymous connection mode, which in itself speaks volume about authentication problems, FTP only relies on username and passwords to authenticate a user.

Usernames and passwords are a poor way of authenticating people, not always by nature – some passwords can be made fairly complicated to guess or to attack if you take the pain to make them so – but because they are used by humans, who, as we all know, have limited abilities in remembering long and complex passwords and tend to use fairly standards elements such as birthday dates, relatives or pets names when composing their passwords.

Another authentication problem with username and passwords is that this type of authentication fundamentally assumes that whoever enters the credentials is the person to whom the access as been granted. That in itself, like any assumption in the domain of Security, is an even greater risk than a poor password.

Finally, the other problem with FTP authentication mechanisms is that it provides absolutely no authentication of the server side. That’s right, why should you be providing your credentials to a server if you can’t even be sure that you’re really communicating with a valid FTP server and not with a fraudulent server that has spoofed the IP address of the real one in order to capture your credentials?

Lack of data integrity

A direct consequence of transmitting data in clear text and not properly authenticating both ends of the communication channel is a lack of data integrity. Indeed, the FTP protocol cannot guarantee that the data that’s been transmitted between point A and B has not been altered in any way. This is a major issue especially when transmitting sensitive information.

Because of FTP lack of data integrity mechanism, an attacker could set up a proxy between the FTP client and the FTP server that would relay messages between the two endpoints, making them believe they are talking directly to each other. The attacker would then have all latitude to intercept all messages going between the two victims and alter the data being transmitted.

Firewall-unfriendly

One of the fundamental shortcomings of FTP is its inability to easily traverse firewalls: a potentially significant problem when operating over networks with multiple layers of perimeter security or when providing access from the outside without additional security systems in place.

The reason why FTP is not friendly to firewalls is primarily because it uses two ports to operate, one of them being ephemeral. Port 21, the FTP well-known port, is the one used for command control. This is the port that FTP uses to exchange commands between the client and the server. However, data transmission is done on another port, a random high-number port that is chosen by the client and transmitted to the server.

To allow FTP transmission through a firewall, a large range of ports must be left open (such as those over port 1024). The firewall is then susceptible to port scans and the vulnerabilities they represent.

Other vulnerabilities

There is a significant number of other vulnerabilities that FTP could be subject to however discussing them in this paper would be an overkill. All of these vulnerabilities are well documented on the Internet and those who are interested in learning more about them will have no problem finding the information. Many of them are related to specific implementations of the FTP protocol while others are general. A few examples are:

- The glob vulnerability: a mishandling of a particular character string that would not return a proper error condition and free memory containing user supplied data allowing an attacker to execute arbitrary code with the system privileges of the FTP daemon
- The bounce attack: an exploit of the FTP protocol whereby an attacker is able to use the PORT command to request access to ports indirectly through the use of the victim machine as a middle man for the request
- Many vendor-specific exploits can also be found online on exploit tracking sites such as the CERT (Computer Emergency Readiness Team), Securityfocus, Secunia or Packetstorm to name a few.

Achieving Secure File Transfer

Open Text Security Family

The Open Text Connectivity Solutions Group, has released a cost effective, yet powerful, suite of software that helps organizations tackle their file transfer security challenges. This suite of software, known as the Open Text Security Family, is composed of three products:

- Open Text Secure Shell
- Open Text Secure Terminal
- Open Text Secure Server

While Open Text Secure Shell and Open Text Secure Terminal are meant to be used as secure file transfer clients, Open Text Secure Server allows organizations to provide their users a high-performance windows-based file server.

Open Text Secure Shell

Open Text Secure Shell is a windows based secure shell add-on for other Open Text Connectivity software such as Exceed®, the world leading PC X server, and HostExplorer®, Open Text's PC-to-host and Web-to-Host terminal emulation software. Open Text Secure Shell has earned the Compatible with Windows 7 and Citrix Ready logos.

Open Text Secure Terminal

Open Text Secure Terminal is a secure shell client for Windows environments. It provides a full-fledged web-based and desktop-based secured terminal and file transfer facility as well as network communication encryption capabilities.

Open Text Secure Server

Open Text Secure Server offers a high-performance highly scalable replacement for Telnet and FTP servers, providing organizations with the same flexibility and convenience than these protocols but in a much more secure environment.

FTP vs. SFTP

All members of the Security family of product are built on the Secure Shell protocol, a robust network transport that offers many advantages over less secure protocols. When it comes to file transfers, Secure Shell offers a powerful alternative to FTP through its SFTP protocol. Below is a side-by-side comparison of key security aspects of FTP vs. SFTP.

Category	FTP	SFTP
Transmission of data	Data is transmitted in clear text, allowing an attacker to eavesdrop on the network and intercept the communication's content	Data is fully encrypted using robust algorithms such as AES and transmitted over a secure channel established between the client and the server. Attackers eavesdropping on the network will only be seeing encrypted traffic.
Client authentication	Credentials are transmitted in clear text over the network and could be obtained by an attacker using a network sniffer. Authentication is done through usernames and passwords which can potentially be exposed through brute-force attacks, dictionary attacks or social engineering attacks. Anonymous mode available and often turned on by default allowing users to connect, browse, download and sometimes upload data without being required to provide credentials.	Credentials are fully encrypted and use different mechanisms to guarantee protection of the encryption shared secret (the key being used for encrypting and decrypting the data stream) Authentication mechanisms include user names and passwords, public and private keys, digital certificates, Kerberos ticket. Authentication methods are extensible through the use of keyboard interactive authentication or other pluggable modules. Anonymous mode would defeat the purpose of using SFTP and therefore is not implemented on most clients or servers.
Server authentication	No server authentication, attacker could impersonate the server without being noticed.	Server authenticates during initial negotiation with the client by sending its public key.
Data Integrity	No data integrity mechanism allowing attackers to compromise and alter data without being noticed.	Data integrity algorithms that allow both clients and server to protect the content of the transmission and prevent modification of the data.
Firewall transversal	Use of port 21 for protocol control transmissions. Use of a random high-number port for data transmission requiring firewall to be widely open for communication to take place.	All communications (command and data) are done over port 22 with no requirement to open any supplemental port on the firewall.

Quick return on investment

With billions of losses attributed to security breaches each year and 340 millions record compromises in the United States alone between January 2005 and November 2009, there is ample evidence that the costs of recovering from a security breach (estimated at anywhere between $90 and $305 per compromised record) by far outweigh the investment required to set up a secure file transfer environment using the Open Text Security suite of products.

Although each company would have to build its own business case to determine the cost / benefit ratio of a secure file transfer solution, below is a simple example based on a fictitious company case.

Acme Inc is a medium sized company of 400 people who build and sell furniture to consumers through a dozen shops they own in Canada and in the United States. Every night, Acme Inc pulls the transactions of the day from each shop to their headquarters in order to have them processed by accounting. On average, each shop has 40 transactions a week. Each transaction contains customer information such as contact information, social security number (for credit approval purposes) and credit card numbers.

Should the FTP transfer between the shops and the headquarters be intercepted upon or should an attacker gain access to the ACME Inc server by sniffing their network (through an unprotected wireless hotspot for instance), ACME Inc would compromise around 25,000 records for every year they’ve been in business.

According to the estimates made by the Ponemon Institute, a security breach would cost an average of $197 per record compromised. For a company such as ACME Inc, the total amount of the bill could go close to $5M.

Putting in place a secure file transfer mechanism through Open Text Security family of product would not be expensive for ACME Inc. They would need to purchase a Open Text Secure Server license for every shop where the daily transactions are made available and one Open Text Secure Terminal license in their headquarters from where they would drive the collection of data.

Here is a quick summary comparing the TCO of the Open Text Security solution vs. the cost of a potential security breach.

TCO	Acquisition of 12 Open Text Secure Server administrative edition license and 1 Open Text Secure Terminal license	$6000
	Yearly maintenance	$1200
	Labor setup cost 8 hours initial setup (including software learning) + 2 hours per server at $50 an hour	$1600
	Labor support cost: 4 hours per server per year	$2,400
	TOTAL	**$11,200**

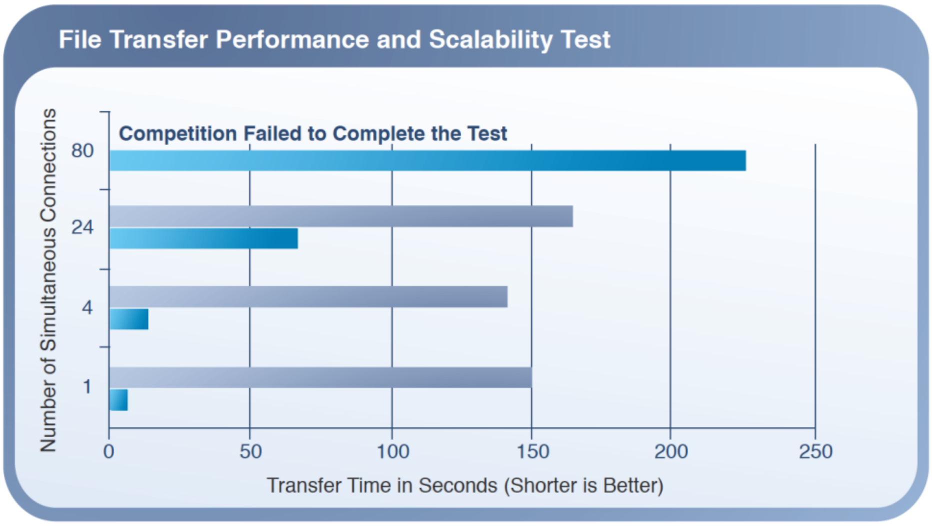
Cost of compromised data in case of security breach	**40 transactions x 12 shops x 52 weeks x $197 per compromised record**	**$4,917,120**

Higher performances and scalability

Although the market abounds with solutions similar to Open Text Secure Server, it remains unbeatable when it comes to performances. Designed by creator of the world renowned Exceed product solutions, Open Text Secure Server can easily blaze through a file transfer task in less than 5% of the time that the competition needs.

In a market where customers are often being forced to choose between security and performance, Open Text Secure Server offers an uncompromising solution that can satisfy security requirements and improve workflow efficiency and user productivity at the same time.

This insurmountable performance advantage is made possible by the state-of-the-art multi-threaded architecture which is thoroughly and meticulously designed to fully exploit the networking characteristics and capabilities of the underlying operating system.



Tolly Group was commissioned to conduct series of test in December 2008. In this round of test, Open Text Secure Server not only outperformed the competition, it could simultaneously support more user sessions and outlasted many similar solutions in the market thanks to the highly scalable design that leaves very small memory and processor footprints. With faster file transfer, better up-time, as well as support for a higher number of simultaneous user sessions, Open Text Secure Server enables organizations of any size to achieve compliance objectives without the burden of additional infrastructure or operational costs. It delivers superior return on investment in mission critical environments.

For more information on the Tolly Group Report, go to:

http://connectivity.opentext.com/resource-centre/other-resources.aspx

Conclusion

Organizations have a lot to lose by continually using FTP for their file transfer needs. With security breaches on an upward trend and their cost rising every year the question that companies have to ask themselves is not “Is FTP a problem?” anymore, but “How much is FTP a problem?”. Stakes are high and the smallest security incident could result in millions of dollars of losses if nothing is done to adequately protect data transfer.

FTP truly is the enemy within. Widely adopted, extremely flexible and easy enough to be used outside of IT control, FTP has gained more ground in the corporate world than most people would admit.

Its inherent security flaws however make FTP a very poor choice to protect data: transmission of data in clear text, weak credentials (if any) relying on usernames and passwords also transmitted in clear text, no mechanism for data integrity, problems with firewall, and several documented exploits against widely used FTP implementations.

The Open Text Security family offers a secure alternative to organizations looking to replace FTP without losing the benefits of its flexibility or dramatically increasing their infrastructure costs.

Open Text Secure Terminal and Open Text Secure Server provide a secure environment where both file transfers are strongly protected with best of breed encryption, robust authentication and real data integrity mechanisms.

By deploying Open Text Secure Server and Open Text Secure Terminal on their network, companies have a cost effective way to protect their mission critical data, comply with industry regulations and satisfy their auditors’ requirements.

About Open Text Connectivity Solutions Group

Open Text's leading Connectivity Solutions connect people, data and applications in mission-critical environments through a complete line of remote application access and data integration solutions. With 90 percent of Global 2000 companies relying on its award-winning solutions for over 20 years, Open Text understands the financial and operational challenges that most organizations face, whether they are multiple systems, disparate data sources, or geographically dispersed teams.

About Open Text

Open Text is a leader in Enterprise Content Management (ECM). With two decades of experience helping organizations overcome the challenges associated with managing and gaining the true value of their business content, Open Text stands unmatched in the market.

Together with our customers and partners, we are truly The Content Experts™, supporting 46,000 organizations and millions of users in 114 countries around the globe. We know how organizations work. We have a keen understanding of how content flows throughout an enterprise, and of the business challenges that organizations face today.

It is this knowledge that gives us our unique ability to develop the richest array of tailored content management applications and solutions in the industry. Our unique and collaborative approach helps us provide guidance so that our customers can effectively address business challenges and leverage content to drive growth, mitigate risk, increase brand equity, automate processes, manage compliance, and generate competitive advantage. Organizations can trust the management of their vital business content to Open Text, The Content Experts.

http://connectivity.opentext.com

Sales: connsales@opentext.com
+1 905 762 6400

To contact our international sales offices:

http://connectivity.opentext.com/contact-us.aspx

www.opentext.com