SDS

Securing and Managing File Transfers

In today's complex networks, organizations are still reliant on legacy protocols to exchange information with their partners. These transfers are critical for companies to conduct business in a modern, interconnected world.

FTP (File Transfer Protocol) is still a very popular and widely-used protocol, even though it is a dated technology. FTP originated in the 1970s to enable file transfers over ARPANET, a precursor to the internet.

FTP is inherently unsecure. Most technicians understand the flaws in native FTP, but many senior executives don't recognize the real risks it poses. Compliance measures like PCI, HIPAA, and SOX are forcing businesses to migrate to secure solutions and adopt newer, safer protocols to avoid severe compliance penalties. Companies need to take a serious look at alternatives that provide more security.

The FTP protocol is still used in environments today despite serious security risks. FTP has potential pitfalls, but alternatives are available without the considerable investment of rewriting applications or major changes to infrastructure. The right technology can make the migration to secure transmission easier and seamless.

The history of FTP

File Transfer Protocol (FTP) was one of the first network applications enabled by TCP/IP. It was developed in 1971 by Abhay Bhushan and standardized with RFC 959 in 1985. Over time, security enhancements were added to make FTP safer. Yet here we are in the 21st century and FTP is alive and well, still widely used despite its inherent flaws.

Why is FTP still being used for data transfers?

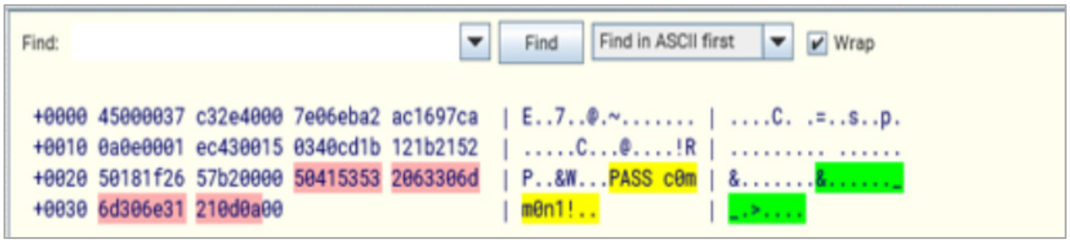
Many institutions use FTP because it is very easy to use. It is the long-standing, de facto standard and, simply put, it "just works." Here are some basic truths about FTP:

- FTP is simple to use.
- Almost everyone has used FTP.
- It is pre-installed on most platforms.
- The command structure is simple.
- It is a well-documented protocol.
- Legacy systems and applications were built with FTP in mind for transmitting files.
- It can be a costly exercise to migrate to a secure solution.
- Any other technology adopted would require re-education.
- FTP is so entrenched in environments that it's difficult to identify who and what is using the FTP protocol.
- And lastly, a tongue-in-cheek remark. The old systems programmer rule: "If it's working, leave it alone."

Why is FTP considered unsafe?

Here is the inside "scoop" on why you should not use FTP in today's modern networking environment.

- FTP is an unsecured protocol.
- It is vulnerable to man-in-the-middle attacks.
- It lacks data integrity.
- It is not firewall friendly.
- Data and passwords are transmitted in plain text, as shown in this packet trace:



```
+0000 45000037 c32e4000 7e06eba2 ac1697ca  | E..7..@.~....... | ....C. .=..s..p.
+0010 0a0e0001 ec430015 0340cd1b 121b2152  | .....C...@....!R | ......... ......
+0020 50181f26 57b20000 50415353 2063306d  | P..&W...PASS c0m | &.......&......_
+0030 6d306e31 210d0a00                    | m0n1!..          | _.>....
```

- Some companies consider their intranet safe enough to use FTP, but a look at the stats on insider threats may change their opinion. *https://www.ibm.com/think/insights/83-percent-organizations-reported-insider-threats-2024*

Should companies still be using FTP as a transfer protocol?

A quick internet search reveals that it is strongly recommended to no longer use FTP due to its significant risks. Cyber breaches are prevalent, and the reputational damage to companies can be catastrophic. *https://www.ibm.com/reports/data-breach*

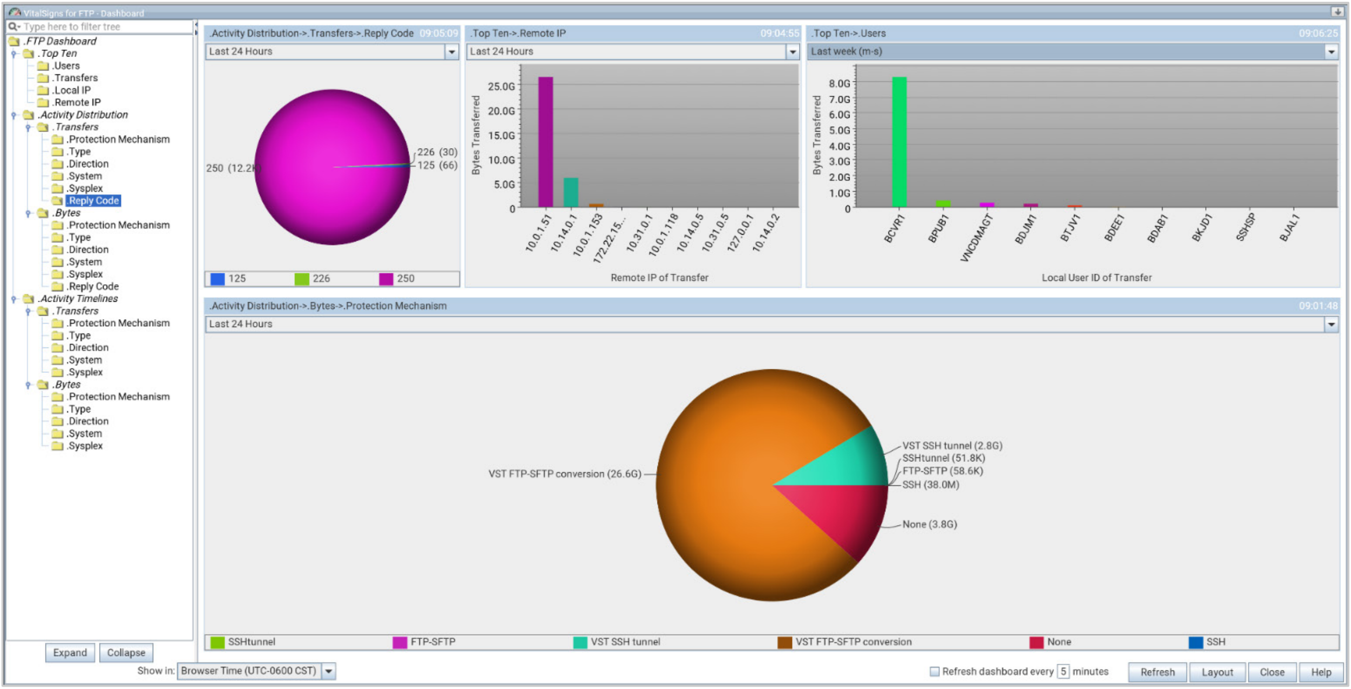
Modern options are available, such as SFTP, FTPS, and cloud-based solutions. If your organization is still using FTP as a transmission mechanism, it may be time to look at alternatives.

Best practices for migrating to a Secure File Transmission Protocol

Step 1 – Visibility: you can't secure what you can't see

- What kind of files are sent and retrieved at your site?
- Can you identify which transfers are secure vs. unsecure?
- Can you identify failing transfers?
- Do you know which FTP scripts are being used?
- Can you identify if users have appropriate access to files on z/OS?
- Are users authorized to download files that they have access to on z/OS?
- Having complete visibility could allow you to safely identify transfers that are still in use and perhaps eliminate redundant transfers, thereby saving CPU cycles.

Do you know all the FTP transfers occurring at your site?



Show in: Browser Time (UTC-0600 CST)

□ Refresh dashboard every 5 minutes

Step 2 – Organization security standards

- Who are you sending files to, and what security requirements must be followed?
- What are the compliance requirements relevant to your industry?
- What level of authentication or ciphers are required for exchanging files?
- What are the audit requirements for retaining files for historical purposes?
- Are you adhering to the security policies relevant to your organization and industry standards?

Step 3 – Implement a secure transfer solution

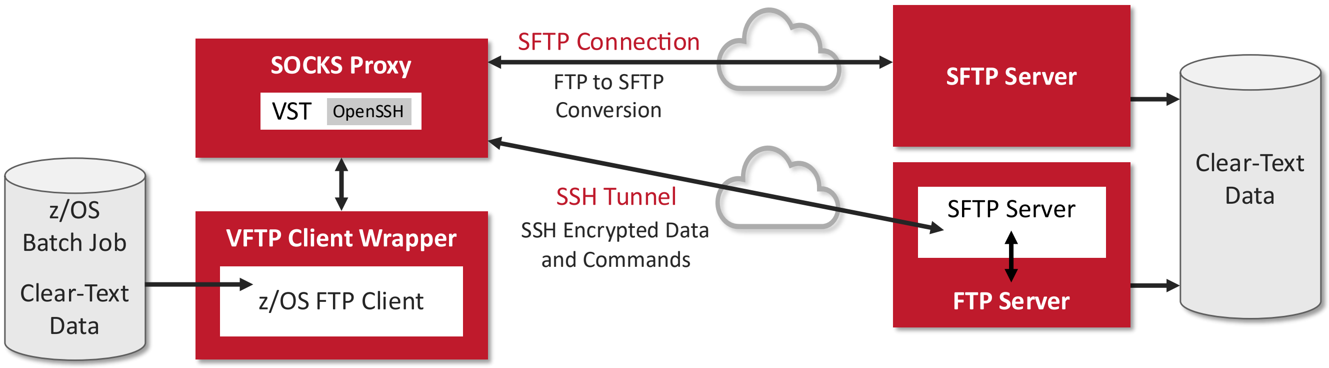
Numerous options are available to migrate FTP to a secure medium. Consider the following before choosing a solution.

Implement a **solution** that:

- best centralizes and manages control of FTP transmissions
- is seamless and minimizes disruption to your organization
- provides central logging of all transfers
- provides complete security and management of all user profiles
- is scalable
- uses a technology that is easy to manage and understand
- is cost-effective
- is easy to install
- has detailed documentation

Work with a **vendor** who:

- understands your requirements
- is open to product enhancements and constructive ideas
- is a leader in secure file transmission technology
- has experience with and understands the z/OS environment
- provides superior support
  Secure transfer solutions can initially be a learning curve for end users and will require some additional help during the rollout.
- provides live technician support
  Users like to speak to real people to solve a problem.



Making migration to SFTP easier with VFTP and VST from SDS

VitalSigns for FTP™ (VFTP) and VitalSigns for Secure Transfer™ (VST), when combined with OpenSSH, make the migration from FTP to SFTP simple and easy.

VFTP provides:

- visibility of all FTP transfers
- automation of batch jobs, e.g. automatic FTP restarts
- auditing and context of FTP transfers
- command security, restricting access to privileged commands
- rules to enforce jobs using the SOCKS proxy

VST:

- VST is a Socks proxy running as a task (STC) on z/OS.
- OpenSSH provides the encryption for VST.
- No additional software is required for encryption, as OpenSSH is shipped with the z/OS base.
- The VFTP client (as opposed to the IBM client) allows users to route batch jobs to the VST SOCKS proxy, where FTP is automatically converted to SFTP.
- No batch JCL needs to be altered when using the VFTP client.
- Users can be very granular about which batch job(s) should be directed to the SOCKS proxy, thereby providing complete flexibility during the migration process.
- SMF reporting on VST/VFTP shows the transfer as a single record on the UI, making it easy to correlate information.
- A complete audit trail of all transfers is logged inside the VST task for complete visibility and automation of routines to be invoked.

Why choose VFTP and VST?

- VFTP provides visibility into FTP activity.
- Information can be extracted from the VFTP database and automatically imported into spreadsheets to illustrate migration information and statistics.
- Easier and faster transition than performing a JCL re-write to native SFTP command syntax.
- VST leverages Open SSH encryption software already included on z/OS. No additional software is needed.
- SDS offers hands-on assistance and free training on the new technology until users are comfortable with migration.
- SDS has assisted many customers with their migrations and are experts in this field.
- SDS has award-winning, world-class customer support.

Summary

FTP technology is robust and enduring, but it fails to meet all the requirements for modern business enterprises. SFTP is significantly more secure because it encrypts all data in transit, including passwords and user data.

Using FTP exposes sensitive data to interception and does not meet current cybersecurity standards. Organizations can no longer afford the risk of a data breach and subsequent reputational damage.

Companies are strongly advised to assess their existing infrastructure and migrate from FTP to a more robust, encrypted solution to mitigate risk and safeguard their operations. The shift from outdated, unencrypted FTP to secure alternatives like SFTP is an essential security upgrade for any enterprise that handles sensitive information.

It is time for companies to spring into action and address the FTP vulnerability.