# SecureFTP for VSE

Connectivity Systems
Product Development
*Don Stoever*

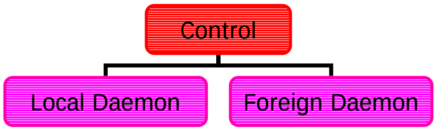
# FTP Protocol

- File Transfer Protocol
- Easy for transferring files between different platforms
- Widely Used
- Old and Reliable
- Efficient but not very secure
- Defined in RFC0959

## FTP Protocol

- What is a protocol?
  - Rules for communicating
  - Command syntax
- Control connection
- Separate and Independent Data Connection
- Control connections sends a command and receives 3 digit numbered reply

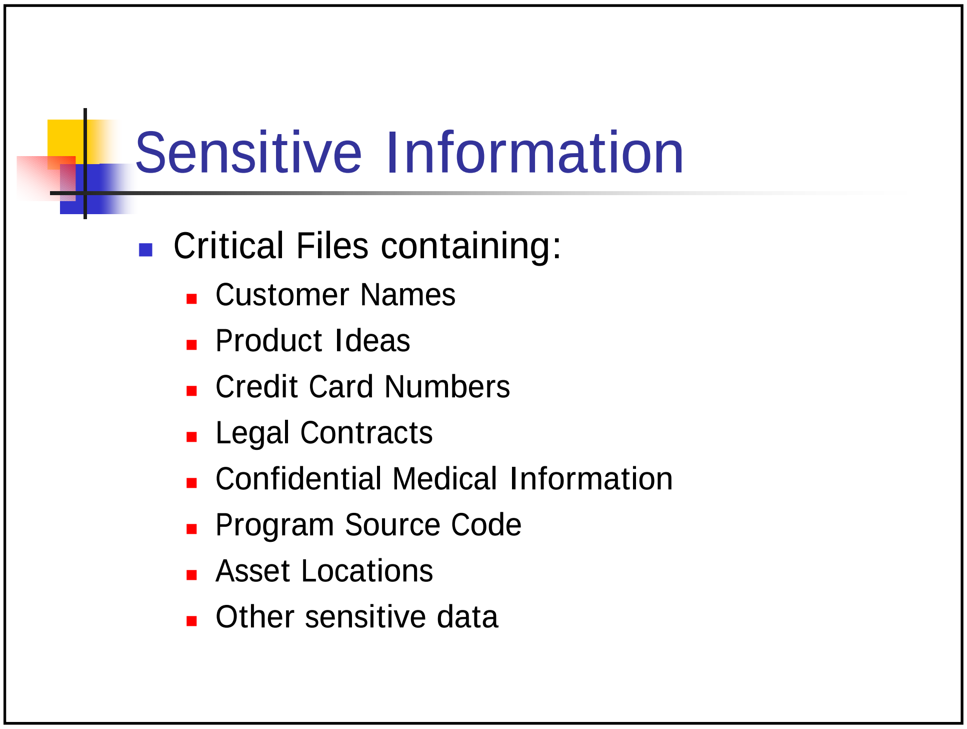
## FTP Protocol



Control

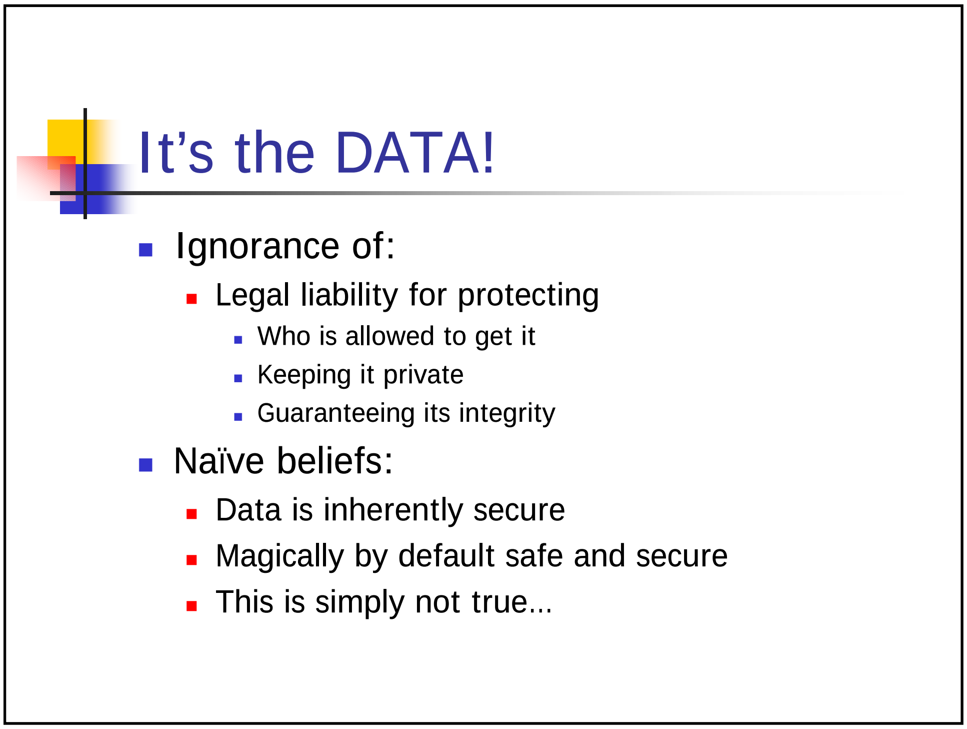
Local Daemon

Foreign Daemon



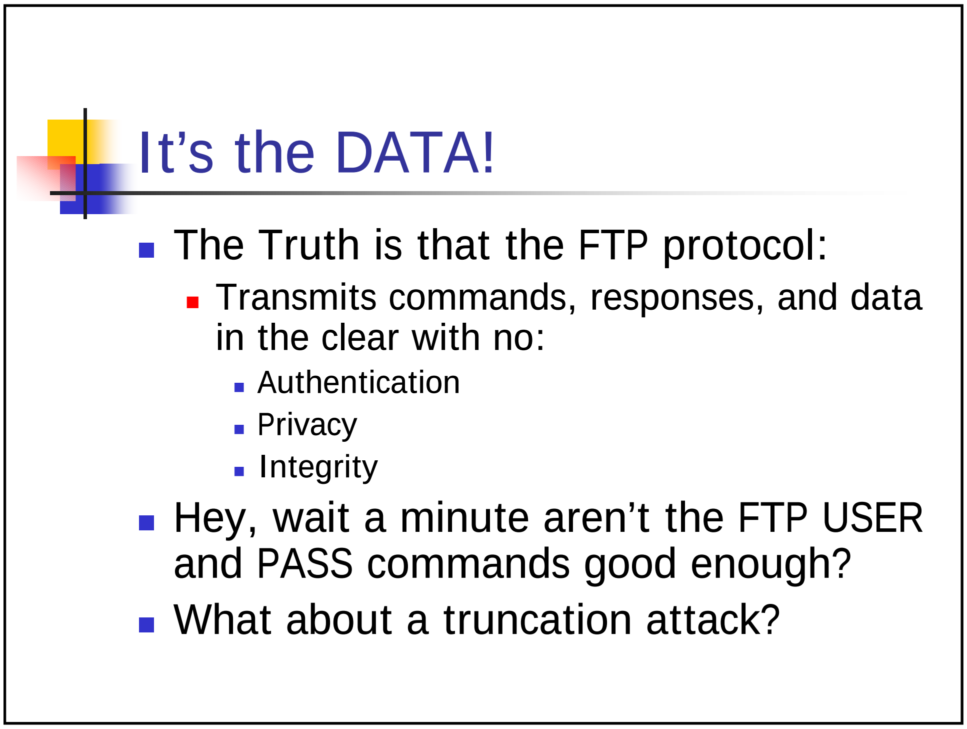
## Sensitive Information

- Critical Files containing:
  - Customer Names
  - Product Ideas
  - Credit Card Numbers
  - Legal Contracts
  - Confidential Medical Information
  - Program Source Code
  - Asset Locations
  - Other sensitive data



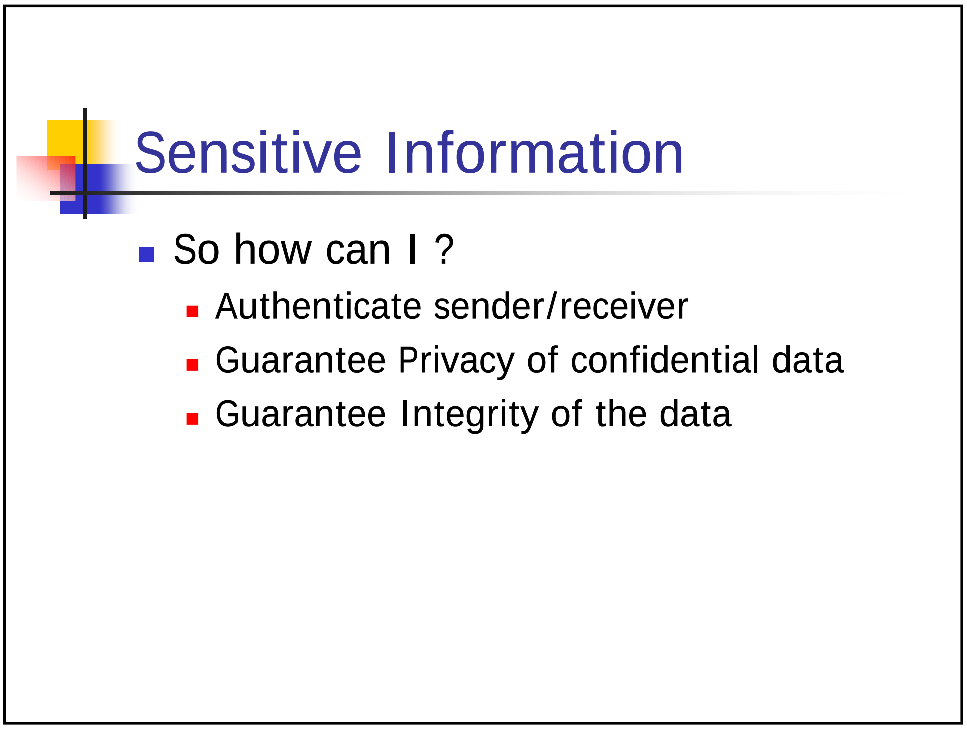
## It’s the DATA!

- Ignorance of:
  - Legal liability for protecting
    - Who is allowed to get it
    - Keeping it private
    - Guaranteeing its integrity
- Naïve beliefs:
  - Data is inherently secure
  - Magically by default safe and secure
  - This is simply not true…



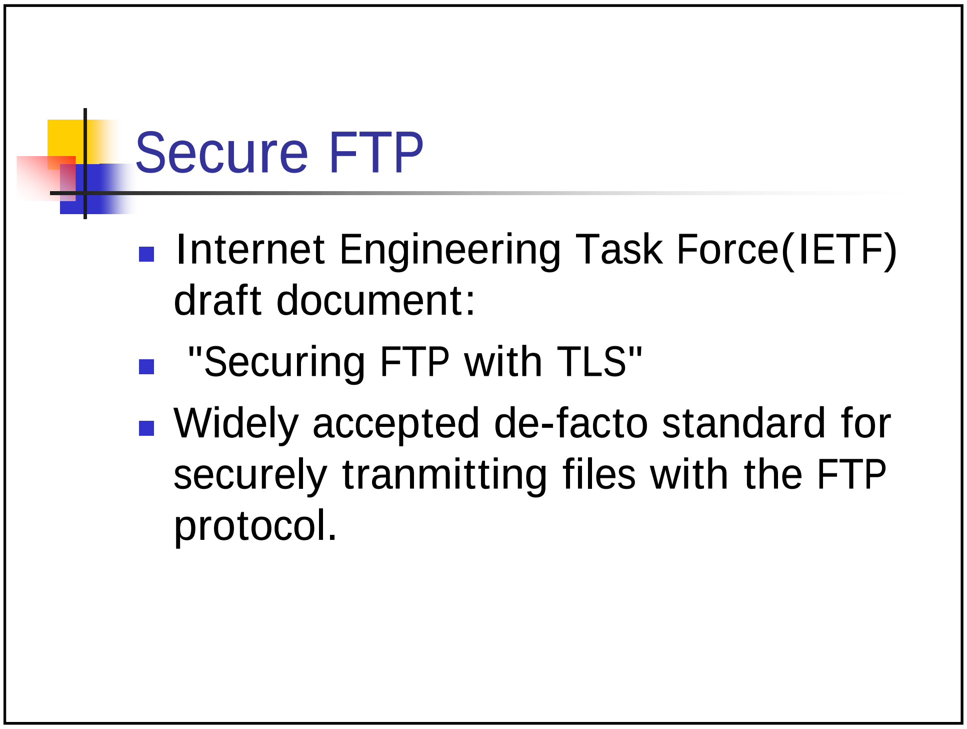
## It’s the DATA!

- The Truth is that the FTP protocol:
  - Transmits commands, responses, and data in the clear with no:
    - Authentication
    - Privacy
    - Integrity
- Hey, wait a minute aren’t the FTP USER and PASS commands good enough?
- What about a truncation attack?



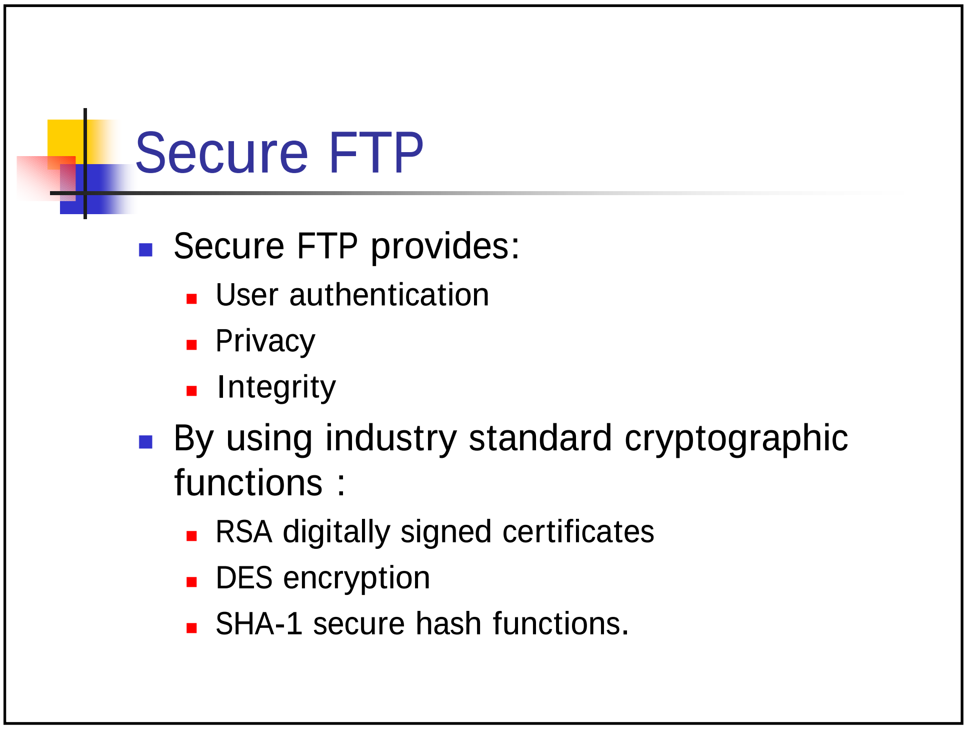
## Sensitive Information

- So how can I ?
  - Authenticate sender/receiver
  - Guarantee Privacy of confidential data
  - Guarantee Integrity of the data



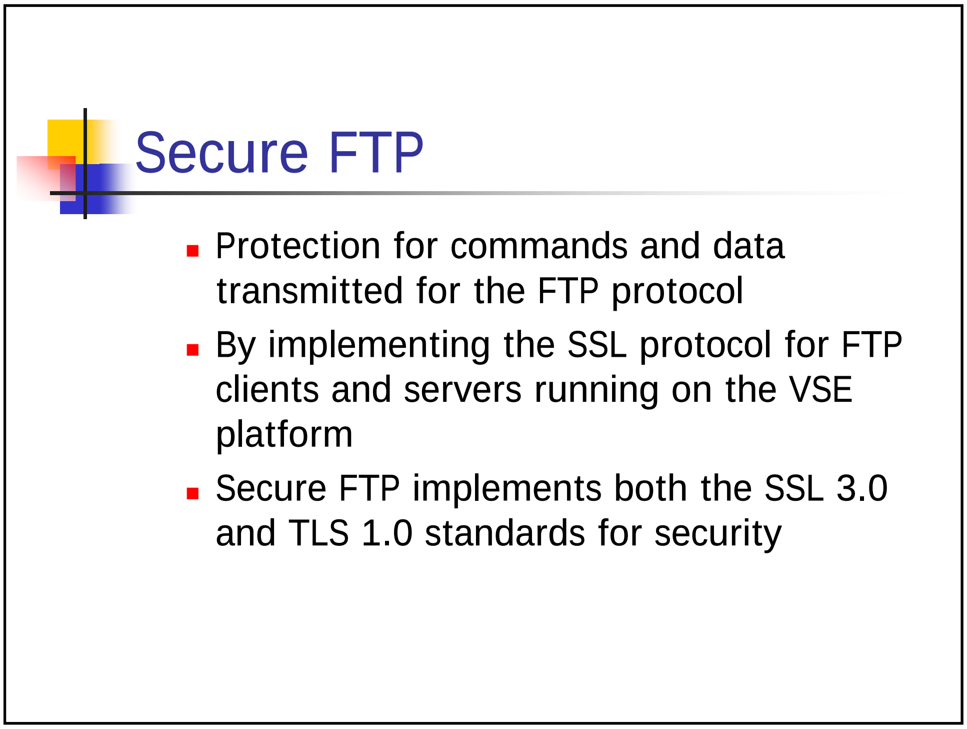
## Secure FTP

- Internet Engineering Task Force(IETF) draft document:
- "Securing FTP with TLS"
- Widely accepted de-facto standard for securely tranmitting files with the FTP protocol.



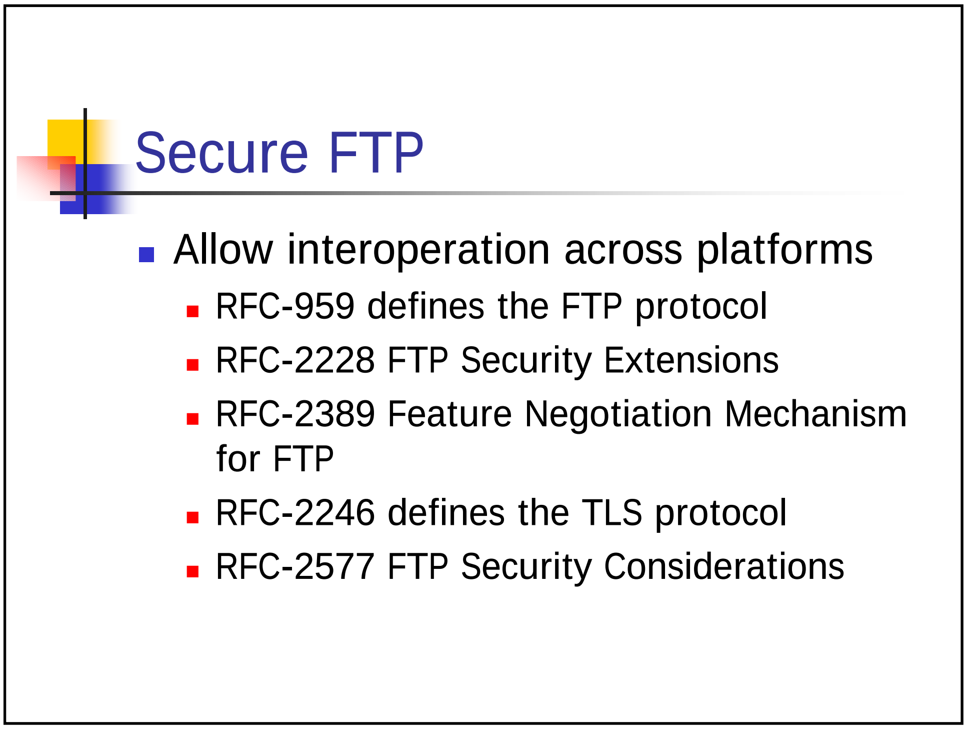
## Secure FTP

- Secure FTP provides:
  - User authentication
  - Privacy
  - Integrity
- By using industry standard cryptographic functions :
  - RSA digitally signed certificates
  - DES encryption
  - SHA-1 secure hash functions.



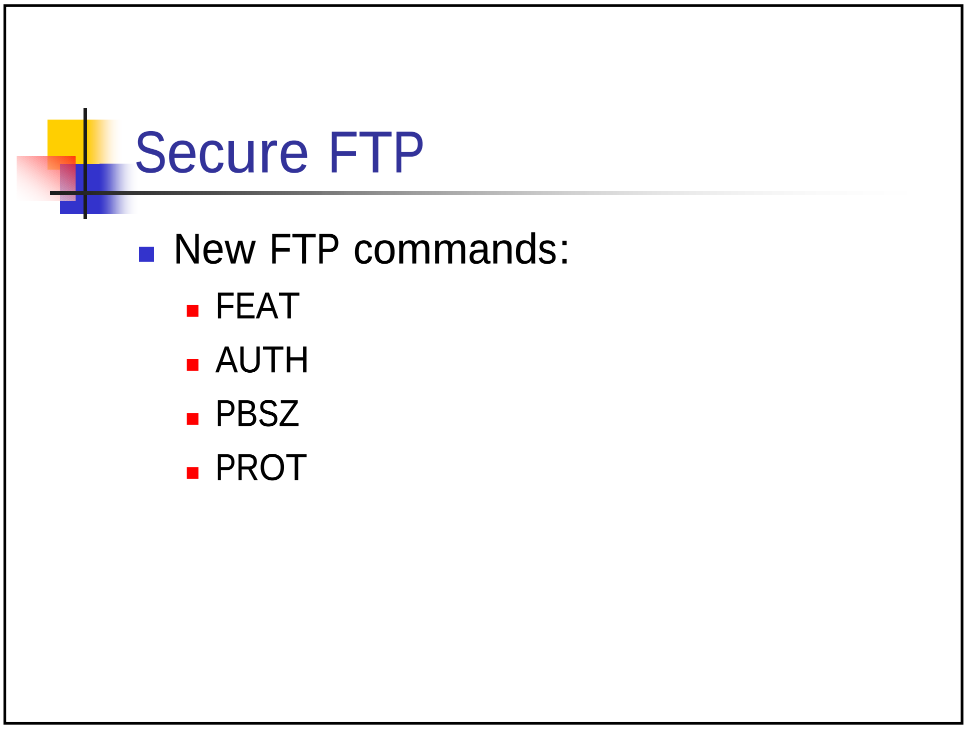
## Secure FTP

- Protection for commands and data transmitted for the FTP protocol
- By implementing the SSL protocol for FTP clients and servers running on the VSE platform
- Secure FTP implements both the SSL 3.0 and TLS 1.0 standards for security



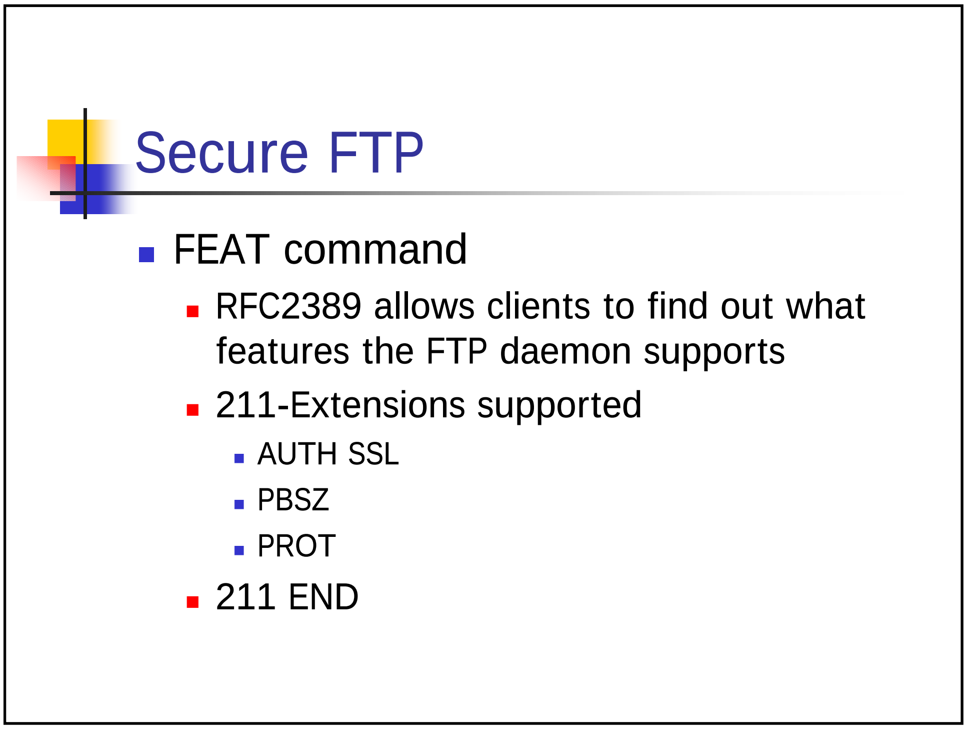
## Secure FTP

- Allow interoperation across platforms
  - RFC-959 defines the FTP protocol
  - RFC-2228 FTP Security Extensions
  - RFC-2389 Feature Negotiation Mechanism for FTP
  - RFC-2246 defines the TLS protocol
  - RFC-2577 FTP Security Considerations



## Secure FTP

- New FTP commands:
  - FEAT
  - AUTH
  - PBSZ
  - PROT



## Secure FTP

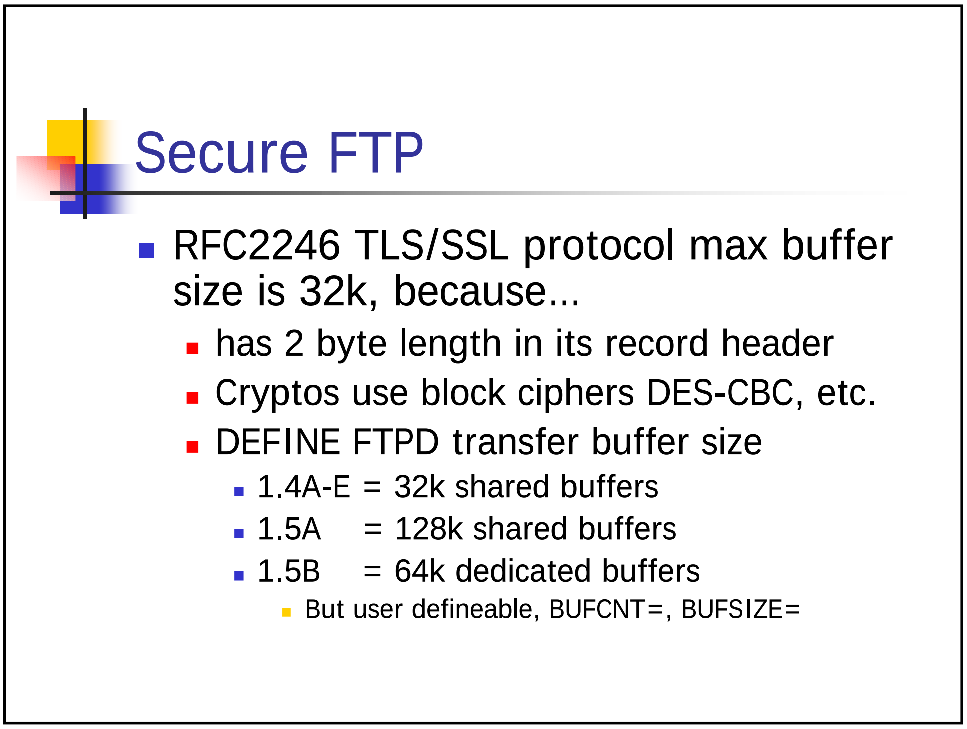
- FEAT command
  - RFC2389 allows clients to find out what features the FTP daemon supports
  - 211-Extensions supported
    - AUTH SSL
    - PBSZ
    - PROT
  - 211 END

## Secure FTP

- AUTH SSL command
  - Issued by client
  - Causes a SSL session to be negotiated
  - Must be first command after OPEN
    - All other commands rejected until SSL enabled FTP daemon gets this!
  - Protects the control/command connection to the foreign FTP daemon
  - AUTH TLS also allowed as synonym
  - SSL is self-negotiating…

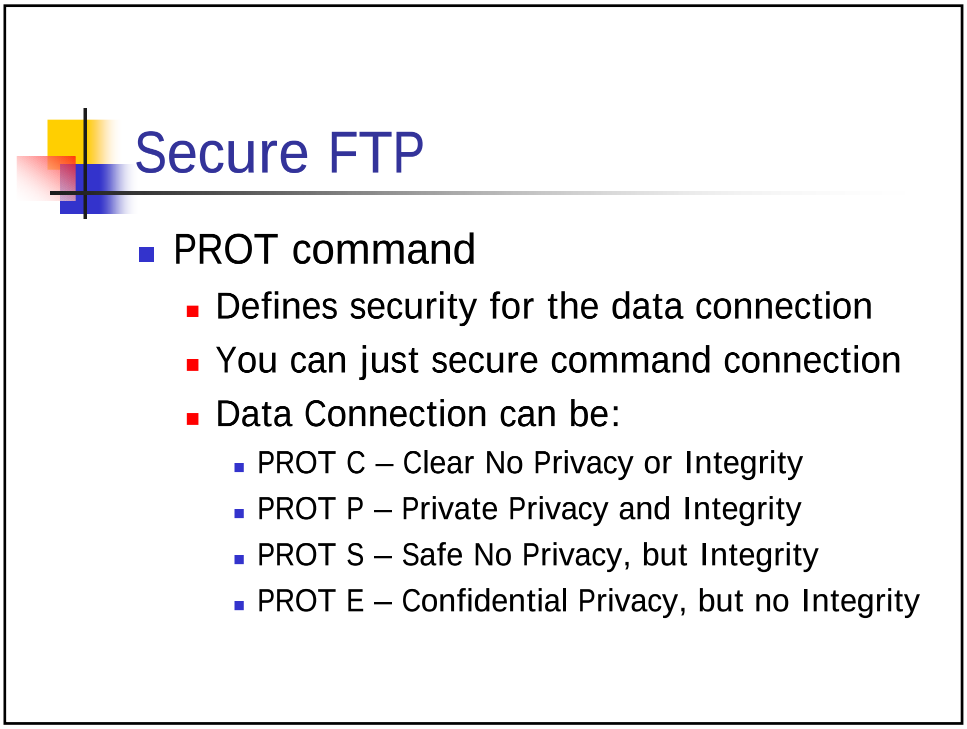
## Secure FTP

- PBSZ command
  - RFC2228 Protection Buffer Size
  - Required Prior to PROT command
  - Not coded by end-user
    - Like when you do a PUT, internally FTP issues PORT, RETR, STOR
  - Not really used for anything but is still required…because…



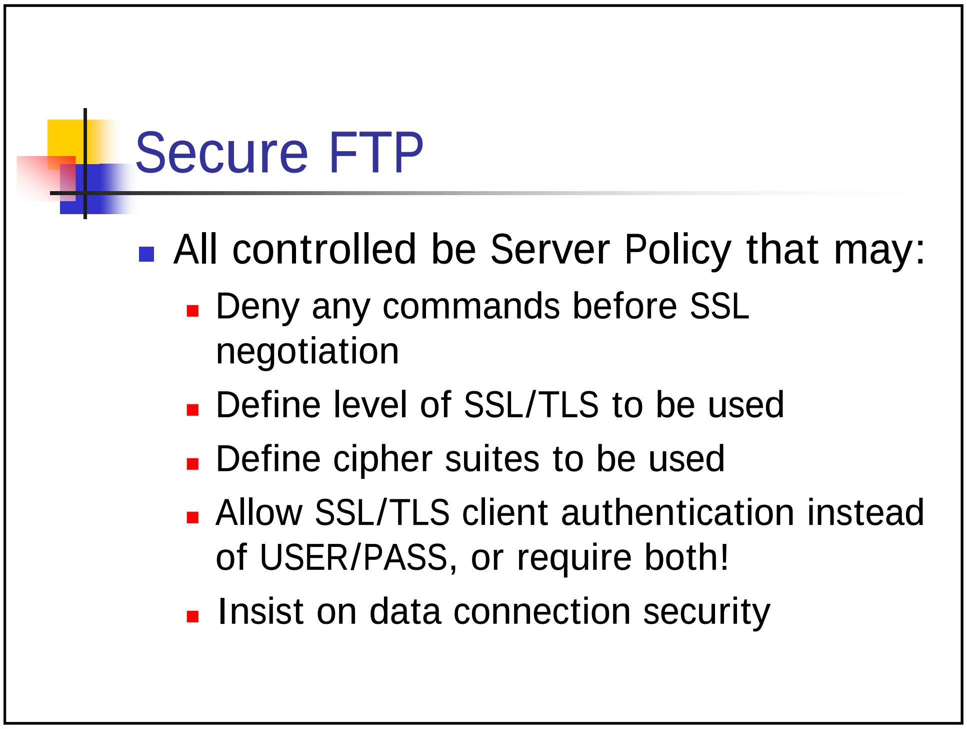
## Secure FTP

- RFC2246 TLS/SSL protocol max buffer size is 32k, because…
  - has 2 byte length in its record header
  - Cryptos use block ciphers DES-CBC, etc.
  - DEFINE FTPD transfer buffer size
    - 1.4A-E = 32k shared buffers
    - 1.5A = 128k shared buffers
    - 1.5B = 64k dedicated buffers
      - But user defineable, BUFCNT=, BUFSIZE=



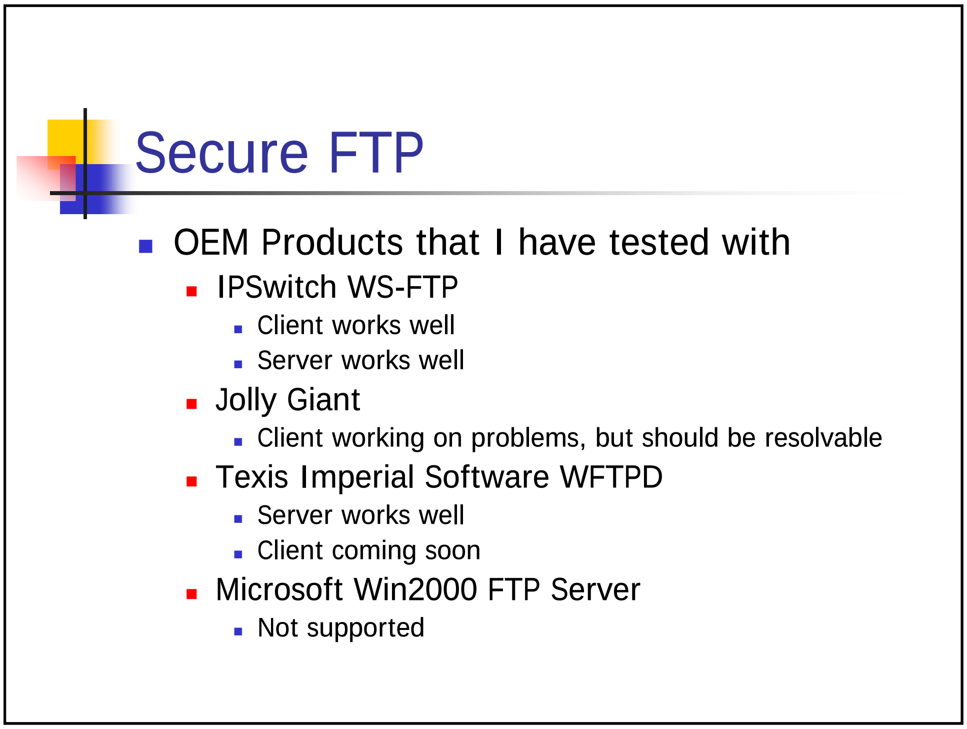
## Secure FTP

- PROT command
  - Defines security for the data connection
  - You can just secure command connection
  - Data Connection can be:
    - PROT C – Clear No Privacy or Integrity
    - PROT P – Private Privacy and Integrity
    - PROT S – Safe No Privacy, but Integrity
    - PROT E – Confidential Privacy, but no Integrity



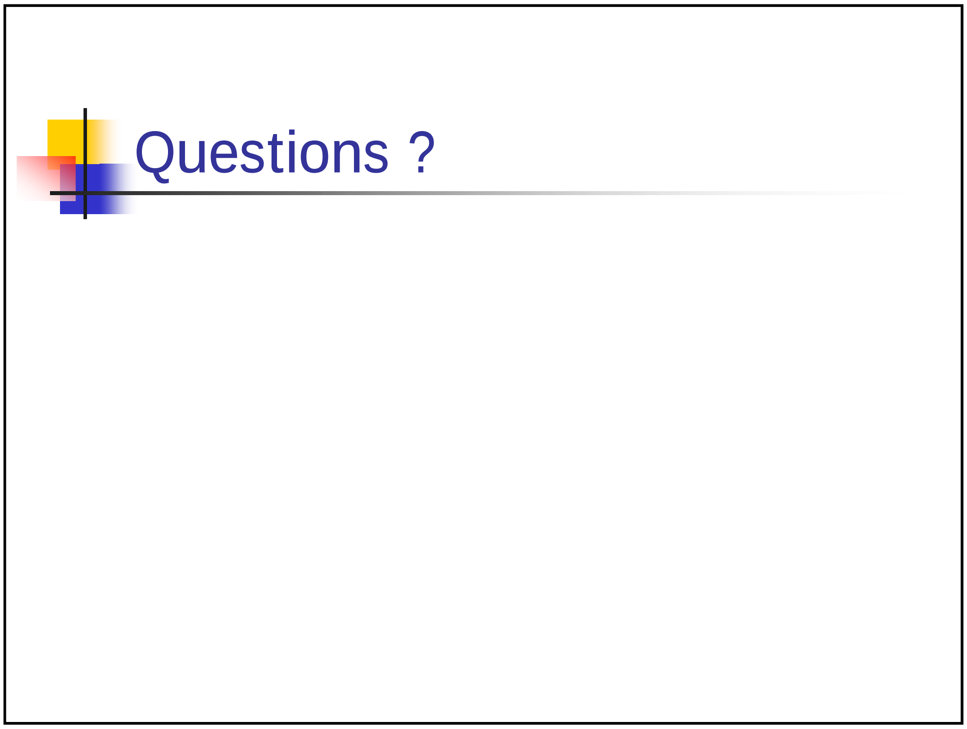
## Secure FTP

- All controlled be Server Policy that may:
  - Deny any commands before SSL negotiation
  - Define level of SSL/TLS to be used
  - Define cipher suites to be used
  - Allow SSL/TLS client authentication instead of USER/PASS, or require both!
  - Insist on data connection security



## Secure FTP

- OEM Products that I have tested with
  - IPSwitch WS-FTP
    - Client works well
    - Server works well
  - Jolly Giant
    - Client working on problems, but should be resolvable
  - Texis Imperial Software WFTPD
    - Server works well
    - Client coming soon
  - Microsoft Win2000 FTP Server
    - Not supported



## Questions ?