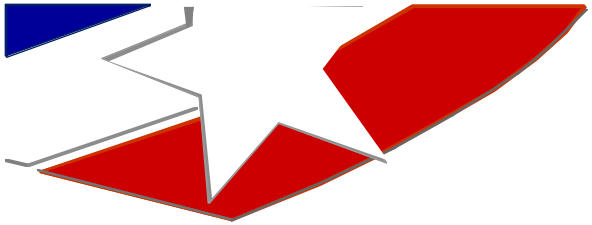


Firewalling Windows NT: A Case Study

May 5, 1999

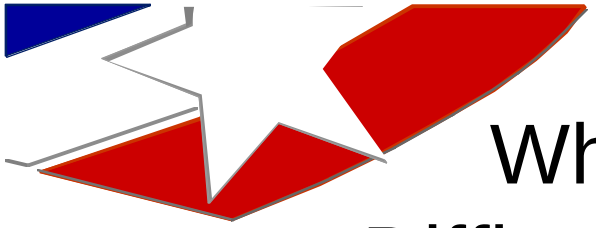
C. Douglas Brown

cdbrown@sandia.gov

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

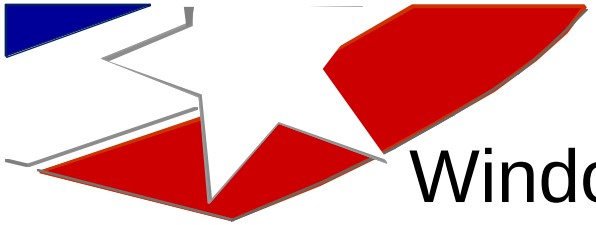
SAND 99-1261C

Sandia National Laboratories



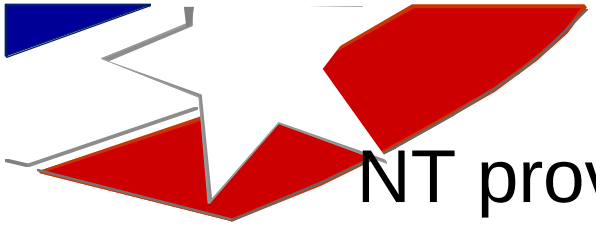
What Makes Windows NT Difficult to Protect via a Firewall?

- In a word, Versatility!
- Services are accessible and easy-to-use
  - multiple communication paths by default
  - can be controlled by binding order
- Backwards compatibility is the default



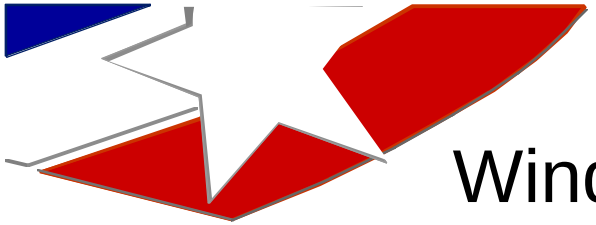
Windows NT is designed to make services accessible and easy to use

- Most capabilities come enabled by default
- Sometimes applications break when the system is tightened down



NT provides backwards compatibility to older, less secure network protocols

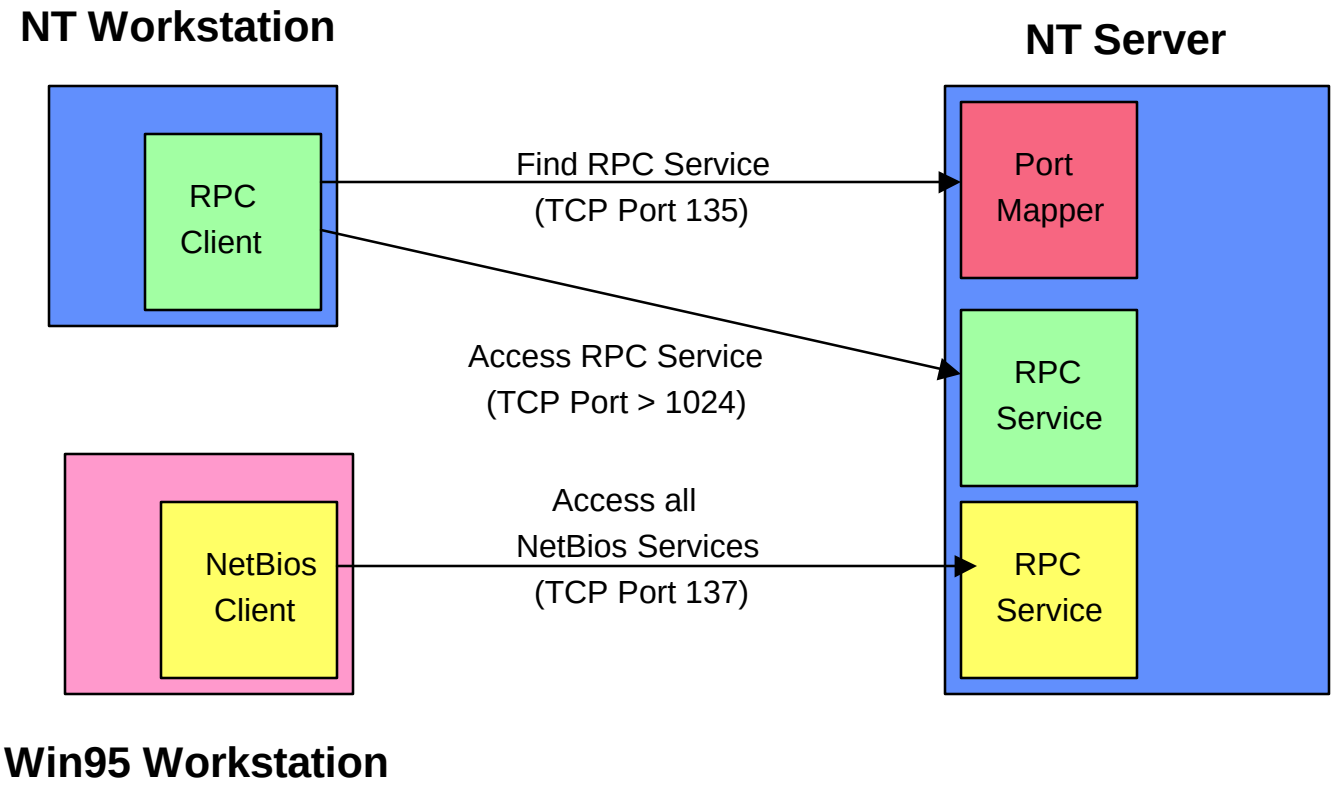
- **Supports access from Win95 running LanManager protocol over NetBios**
  - can be disabled in Service Pack 3, but few do it
- **All NetBios access is tunneled over TCP port 137**
  - no way to selectively control NetBios access
  - stuck with an “all or nothing” decision



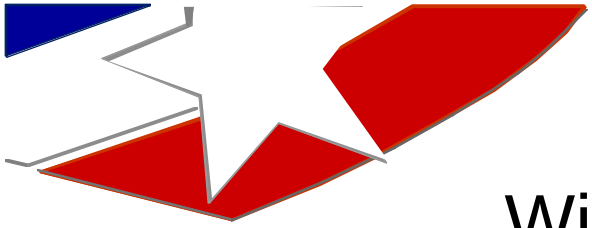
Windows NT can use RPC's to access many network services

- **DCOM uses Remote Procedure Call (RPC)**
  - many applications are DCOM-aware, including MS Office Suite
- **RPC services generally share the same TCP/UCP ports**
  - some Microsoft applications allow TCP/UDP port to be specified
  - can restrict RPC's to specified TCP/UDP port range
- **How is this different from DCE?**
  - can restrict RPC's to specified TCP/UDP port range
  - few DCE services are enabled by default
  - DCE has better authentication mechanism
- **If you permit MS and DCE RPC's through firewall, you could get all RPC services**
  - can use wrappers

Diagram of NT Network Access



Sandia National Laboratories



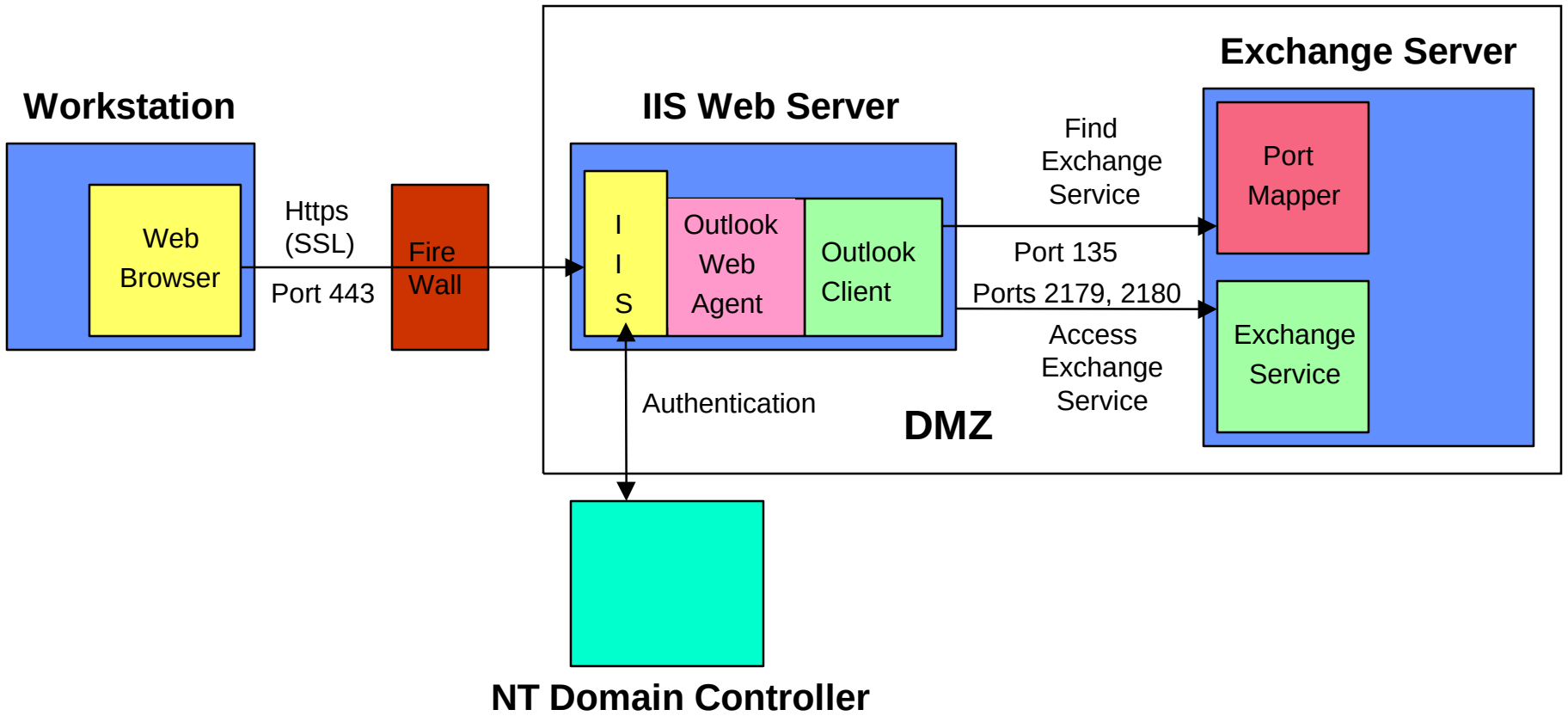
Windows NT Authentication

- **Uses DES encryption**
  - encryption algorithm is same as Kerberos/DCE, but
  - encryption implementation options are weaker
- **Cracker tools exist brute force password attacks**
  - faster for NT than for Kerberos
  - depends upon backward compatibility options
  - takes only a few hours (or less), depending on password strength

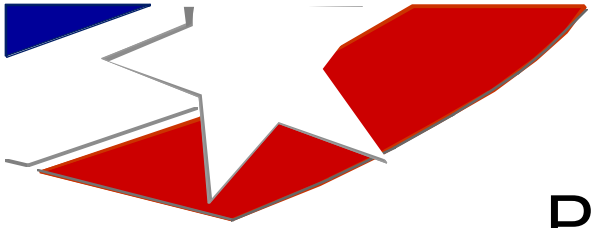
Access to the Microsoft Exchange Server

- **RPC**
  - from Outlook native client only
- **IMAP**
  - from Netscape Communicator or Outlook native client
  - can run over SSL for increased security
  - limited functionality, but better than HTTP
- **HTTP**
  - from any web browser
  - can run over SSL via HTTPS
  - limited functionality, but useable
  - requires more server resources (~6x)

Sandia Phase I Configuration

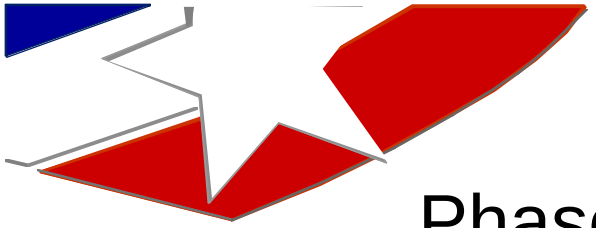


Sandia National Laboratories



Phase I: HTTP over SSL

- **Dedicated web server front end to Exchange**
  - OS configuration of web server may be much tighter
  - can eliminate many unnecessary services
- **Outlook Web Agent runs as proxy application**
  - accesses Exchange in behalf of the user
  - translates data from Exchange into format for HTTP display
  - limited functionality, must be updated as Exchange server is modified / enhanced
- **SSL encryption protects the password and data**
  - but password still might be compromised on client workstation



Phase I: HTTP over SSL (cont.)

- **Considered user authentication via client certificates**
  - certificate must be validated in order to establish SSL tunnel
  - username / password still required for user authentication
  - but, password used to unlock private key could be compromised
  - would require support of a Microsoft certificate authority (CA)
  - could not use existing Entrust CA (maybe future?)
- **Phase I worked and was reasonably secure, but …**
  - management and users demanded more
  - expected full functionality of Outlook native client from outside the firewall (early version of OWA didn't support calendar)

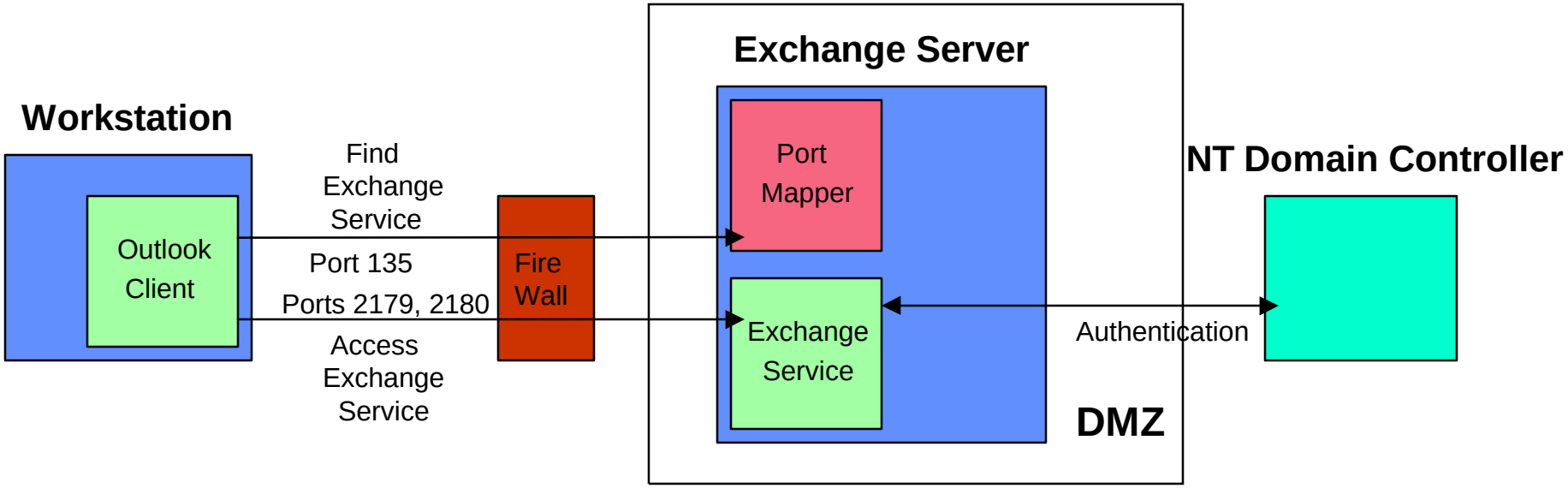
Phase II: RPC-based Access from Outlook

- **The good news: Exchange services can be locked down to three TCP ports**
  - Port 135 -- Port Mapper is contacted to find ports for services
  - Ports 2179, 2180 -- used by Exchange services
  - Can limit access through firewall to Exchange servers only on these ports
- **The problem: RPC's are a general communication mechanism used for many network services**
  - the authentication tokens are encrypted but still somewhat weak
  - cannot force encrypted RPC's from the server, only from client
  - not much experience with security of RPC's -- are there buffer overflow problems, etc.?

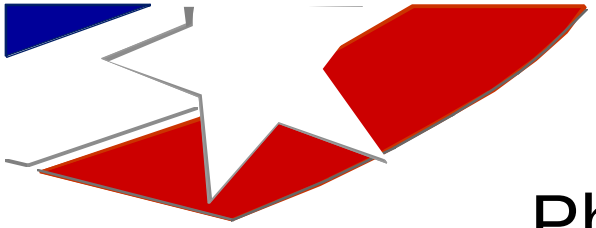
Phase II: RPC-based Access from Outlook

- **Again, client certificates would help**
  - but rejected for same reasons as for HTTP
- **Arranged for a quick literature search on the web for RPC vulnerabilities**
  - none uncovered, but...
  - report expressed concerns with architecture (e.g., exposing RPC)
- **Presented results to management**
  - explained concerns
  - no known vulnerabilities identified
  - received direction to proceed

Sandia Phase II Configuration

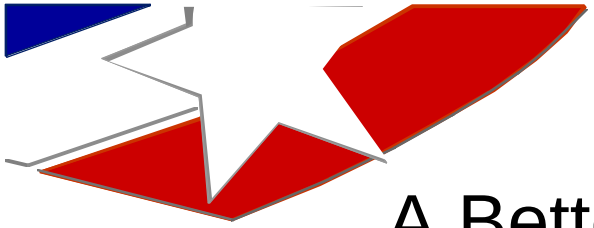


Sandia National Laboratories



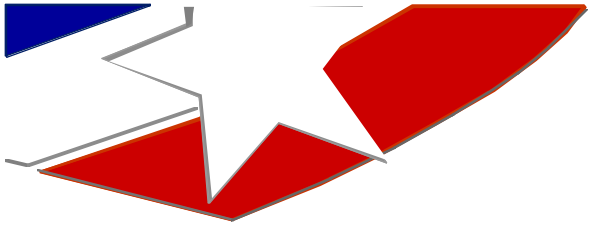
Phase III: IMAP over SSL

- **Examined IMAP4 protocol specifications**
  - protocol has very limited set of commands
- **Queried vendor regarding possible protocol extensions**
- **Configured IMAP4 to run over SSL**
- **Ran ISS Scanner against the system**
  - only proved system wasn't listening on other ports
- **No further testing**
  - configuration was assumed to be at least as secure as RPC



A Better Approach: An RPC Proxy

- **Could enforce secure RPC's**
  - limit access to specific RPC services
  - require authenticated RPC's
  - require encrypted RPC's
- **No commercial RPC Proxy firewalls exist**
  - talking with DASCOM about an RPC Proxy for DCE
  - might be adaptable to Windows NT RPC's



The Bottom Line: Firewalling NT Services is Difficult

- **RPC-based services can be difficult to control**
  - some services can't be locked to specific ports
  - some services don't allow server to require encrypted RPC's
  - the Port Mapper must be exposed to the outside
- **Windows NT comes with many services enabled**
  - hard to find and disable unneeded services
- **A web-based front-end to Exchange is the most secure solution**
  - if possible hold the line there!

SAND 99-1261C