

ProDiscover[®] (Advanced) Architecture, Remote Agent, and Perl

Christopher L.T. Brown, CISSP
Technology Pathways, Founder & CTO
clbrown@techpathways.com
619-435-0906 / 888-894-5500 x111

Copyright © 2005- 2006, Technology Pathways

Purpose of this Presentation

- Review the basic architecture of ProDiscover family of computer forensics products
- Provide an in-depth look at using and troubleshooting the ProDiscover Remote Agent
- Provide attendees a basic understanding of the Perl language
- Introduce attendees to writing, debugging, and running ProScripts

ProDiscover[®] Architecture & Product Family

Tools For The Process

- The ProDiscover[®] family of products are designed to support the forensics process for specific markets
 - ProDiscover[®] – Basic *Freeware*
 - ProDiscover[®] – for Windows
 - ProDiscover[®] – Forensics (includes Perl)
 - ProDiscover[®] – Investigator (includes Perl)
 - ProDiscover[®] – Incident Response (IR) (includes Perl)
 - More to come...
- Basic features and UI are common to all family tools
- Customized functionality in each tool to suit the user

ProDiscover[®] Core Architecture

- Read the disk at the sector level in a Read-Only mode
- Perform all display and functions through it's own trusted, read-only file system
- Currently supports all versions of FAT and NTFS including Dynamic Disk, Software RAID and Volume Sets, SUN Solaris UFS on SPARC and X86 Platforms, LINUX Ext2/3

Refresher Walkthrough

Creating & Saving a New Project, UI,
Help, and User Preferences

Live Imaging & Analysis with ProDiscover[®] *IR*

Why Live Enabled Disk Forensics

- Often the system is live at seizure
- Prevent loss of physical memory contents
- Least intrusive to business operations
- Long term investigations
- Remote artifact extraction
- Cross-over preview & imaging

Benefits of Live/Remote Forensics

- Allows for remote ***Preview & Imaging*** Live systems without taking them off line
 - Pre-search Confirmation
 - Discovery
 - Live Selective Extraction
 - Provisional Warrants (medical, financial, and transactional based systems)
 - Compliance & HR investigations
 - IT Security Incident Response...

Implementation

- ProDiscover[®] *IR* was designed in a client/server model
- ProDiscover[®] - Client or Console
 - Main application functionality
- PDServer[™] - Server or Remote Agent
 - Run on remote system to allow ProDiscover[®] client access to disk

PDServer™ Rem0te Agent

- All Imaging and Preview functions supported
- Agents for all supported platforms
- HPA functions are supported for all remote disks
- Live Network Images are sometimes referred to as a “smear” since bits on the original may change during imaging
- Includes the ability to image **Physical Memory** and remote system’s **BIOS**

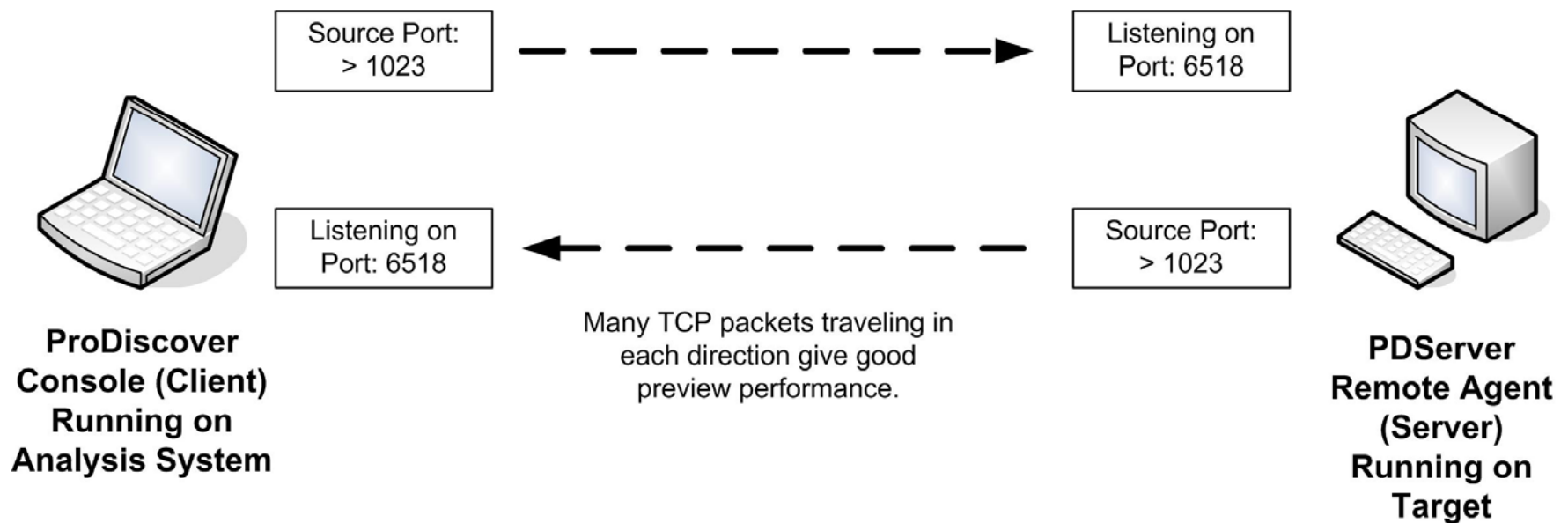
RA Deployment Options

1. PDServer Remote Agent pre-installed
2. Run PDServer Remote Agent from CD-ROM or Thumb-drive
3. GUI “Push” from ProDiscover Console
4. End-point systems management system such as Microsoft[®] SMS.
5. Booting the target system to the PDServer LINUX Boot Disk

Network Topology Dependant

- Understanding the network topology is crucial
 - Routes
 - Connection Speeds
 - Packet Shaping
 - Filters (Access lists on routers, switches & hubs)
 - Firewalls (inbound / outbound filtering)
 - Personal
 - Network
 - NAT, & Port Mapping or PAT

The Remote Connection



Connection Characteristics (1)

- Data channel encryption
 - Unencrypted (default)
 - 256 Bit AES
 - 256 Bit TwoFish
- Remote Agent Visibility
 - Foreground application (default)
 - Stealth mode option
 - Using scripts, menu option or command line
 - PDServer.exe -S:Password
- Server is read only and digitally signed protecting data

Connection Characteristics (2)

- Must be run as root or local system admin
- Configurable port settings for firewall filtering
 - Console set by choosing
 - File | Preferences... | PDServer
 - Remote Agent by Launching from command line
 - PDServer.exe -p:80
- Set remote agent password by command line
 - PDServer.exe -S:password

Connection Characteristics (3)

- GUIDs (Globally Unique Identifiers) are created for each side of the connection which are sequenced to maintain connection integrity and security
- Session setup is always encrypted regardless of user encryption settings
- Bad password lockout algorithm to protect against brute force attacks

Connection Characteristics (4)

- If a session is determined to be “broken” the it must be reset on one of the following
 - Console - “Network | Release Remote Client”
 - Remote Agent - “Tools | Release Client”

Creating Remote Agent

- Windows
 - Windows Remote Agent CD Included
 - Tools Menu | Create Remote Server Disk (uses pdserver.eve file in \Remote Agent\Windows directory)
 - Burn Files from \Remote Agent... directory
- SUN/Linux
 - Burn Files from \Remote Agent\ - Burn ISO image from same directory

Steps to Remote Imaging/Analysis

1. Remote Agent Running on target (push, pre-install etc.)
2. Connect to the target system
3. Add physical disk and/or conduct IR menu functions

Remote Agent on a CD/Thumb Drive

The easy way

Least Intrusive IR Method

- Place Remote Agent in CDROM / USB Drive
 - Auto Run or Run “<Drive>:\PDServer.exe
- Personal Firewall may ask for run authorization
- Runs in memory and locked to physical RAM
- Limited Registry Changes
 - Device Enum & MRU Keys
- Logged in user must be Local Admin/Root

Demo & Lab

Remote Agent on CD

Remote Agent GUI “Push”

Corporate Preferred Method

Remote Agent GUI “Push”

Accessible
from the
Network
Menu

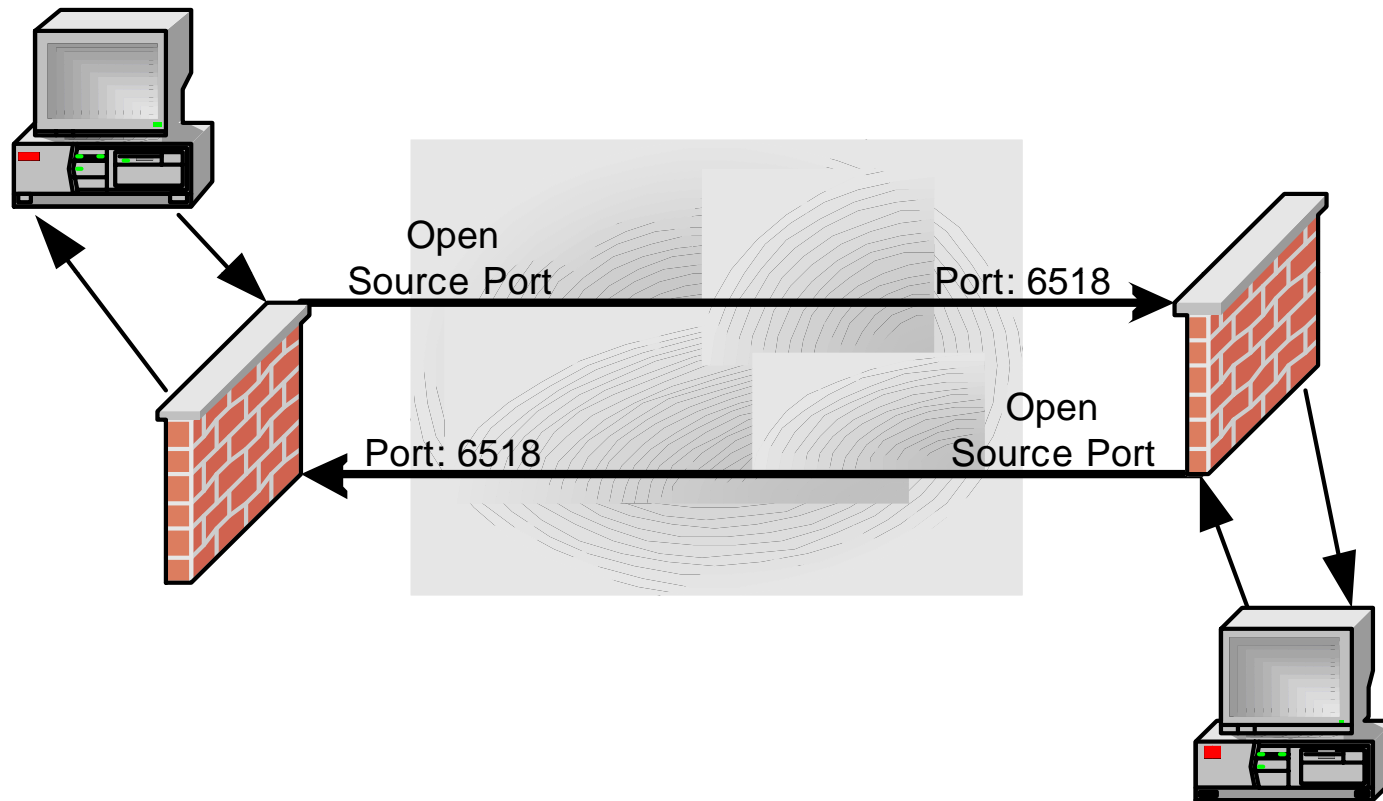
The screenshot shows a Windows-style dialog box titled "Install/Uninstall PDServer on Remote Machine". It contains the following fields and options:

- Enter computer name or IP address: 192.168.100.122
- Radio buttons for:
 - Install PDServer on remote system
 - Uninstall PDServer from remote system
- Installation Type:
 - Install as a Service
 - Install in Windows Run Registry Key
- PDServer settings:
 - Executable Name: PDServer.exe
 - Default Port: 6518
 - Password: [empty text box]
 - Re-enter password: [empty text box]
- Buttons: OK and Cancel

Remote Agent GUI “Push”

- Where are files placed?
 - System32 Directory
- What Files go there?
 - Disk16.dll
 - Disk32.dll
 - msvcrt.dll
 - PDServer.exe
 - DFTSrv.exe (Installed only if installing as a service)
 - DFTSrv.ini (Installed only if installing as a service)
- **Do Not** use an application name already in use by an application in the system32 directory
- During uninstall all files are removed except "msvcrt.dll" which may be required by other applications

Planning & Troubleshooting



The #1 Issue

- The most common issue preventing connections to a remote system is the Windows XP or other personal firewall running on the Remote Agent or ProDiscover console system

Remote Agent “Push” Config (1)

- Security options policy for “Network access: Sharing and security model for local accounts” set for “**Classic – local users authenticate as themselves**” (The default setting is “**Guest only – local users authenticate as Guest**”).
 - Documented in detail on the ProDiscover Community forums at <http://toorcon.techpathways.com/CS/forums/98/ShowPost.aspx>
- Security options policy for “Accounts: Limit local account use of blank passwords to console logon only” is set for “**Disabled**” (The default setting is “**Enabled**”).
 - Documented in detail on the ProDiscover Community forums at <http://toorcon.techpathways.com/CS/forums/99/ShowPost.aspx>

Remote Agent “Push” Config (2)

- Basic Connectivity OK (Ping & Traceroute)
- Windows networking needs to be working
 - Ports (137, 138, 139, 445)
 - Windows XP Firewall “File and Print” enabled under exceptions
 - If remote registry services are not turned on the target system we will turn them on during push.
- Tip: If a failure occurs quickly the problem is most likely firewalling on the target system. Slow failures are usually caused by console firewalling

Planning

- Is the remote approach reasonable to begin with?
 - What is the end-to-end link speed?
 - 100 GB Disk Full Bit-Stream will take > 150 Hours in a perfect point to point T-I
 - Weakest link rule applies (point A with T1 and Point B with T-I does not necessarily = T1)
 - Links that will not work well for full imaging, may work well for preview and selective extraction
 - Understand and test the topology (Basic Connectivity Tests like Ping & Traceroute)

Connection Checklist (1)

- Can you ping the target system? Remember that Ping blocking could be on the client, server, or the network and failure does not definitively indicate a problem.
- If you changed ports on the remote agent, did you change the User preferences on the ProDiscover Console to match?
- Does the remote system have a firewall? If so, is it set to allow the ProDiscover/PDServer port through (6518 by default)?

Connection Checklist (2)

- Does the local system have a firewall? If so, is it set to allow the ProDiscover/PDServer port through (6518 by default)?
- If Windows XP Firewall is installed it is recommended to be turned **ON** at both analysis console and remote target systems with exceptions made for the ProDiscover Port in use (6518 by default). Note Windows XP and other personal firewalls have been known to enforce port blocking when turned **OFF**.
- Is there a hardware firewall or are access lists being enforced on a switch in-between the analysis console and target system? If so the port in use (6518 by default) should be allowed in both directions.

Demo & Lab

Remote Agent GUI “Push”

You're Connected - Now What?

IR Menu Features

Remote Investigations

- All the standard disk features are available
 - Browse the file system
 - Recovered files
 - Select evidence of interest
 - Extract file or groups of files
 - Search
- All functions are slower (link dependant), but available

Menu Items

- Find Unseen Processes
- Find Unseen Files
- Create Baseline
- Compare Baseline
- Find Suspect Files
- Get Process List
- Get System State
- Open/Connected IP Ports

Find Unseen Processes

- Works on Windows systems only
- Builds file tables from remote system starting at sector level
- Analyzes which files have file system locks
- Makes standard system call to determine which processes are running
- Adds to report all seen and unseen processes

Find Unseen Files

- Builds file tables from remote system starting at sector level
- Creates index from file tables
- Request file index from remote system using standard file I/O (will change high-level folder last access times)
- Compares the differences
- Reports files unseen by users

Baseline & Compare

- Forensic level baseline and compare of remote system
- All files are examined from sector level reads and ProDiscover file system display

Find Suspect Files

- Reads hash files in hashkeeper format
- Compares specified directory structure to input hash file
- Reports matches

Get Process List

- Full path to binary
- Process ID
- Process description
- Process dependant libraries/DLLs

Get System State

- User information
- Local Account Settings
- Local Files Open from the Network
- Local Sessions
- Local Shares
- Running Services
- Mapped Resources
- Local User Accounts
- Locally Seen Computers
- Route Tables
- ARP Cache

Open/Connected Ports

- TCP/IP Endpoints for
 - Listening
 - Connected
 - Wait State
- Corresponding binary and process ID

Linux Boot Disk

Linux Remote Boot CD (1)

- Created to allow imaging on site of a disk-at-rest through network or cross-over cable
- Great for Lab work too
- Linux Boot is based on John Andrews DSL (Damn Small Linux)
 - www.damnsmalllinux.org
- DSL was based on Knoppix
 - www.knoppix.org

Linux Remote Boot CD (2)

- Technology Pathways further changed DSL to:
 - Ensure no local system disk are mounted (remote agent provides raw read-only access) Knoppix will increment the journal count on journaling file systems with mounting even in read-only
 - Not auto mount a LINUX swap partition as Knoppix will
 - Remove all non-essential GUI and applications
 - Auto-run the PDServer Remote Agent

Creating Linux Boot Disk

- LINUX Boot CD with remote agent included in product distribution
- Burn ISO image of LINUX boot CD from the \ProDiscover\Linux Boot Disk\ directory

Lab & Demo

Linux Boot CD

Scripting

Introduced Scripting in v. 4.0

- Why Script
 - Provides the power to automate repetitive tasks
 - Standardize Investigations
 - Low level analysis (data carving)
 - Integrate with other tools
 - New data analysis (latest version of ???)

ProScript & Perl

- How is scripting implemented:
 - By exposing low level functions of ProDiscover in the form of the ProScript API (around 150 functions)
 - Functions are made available to ActiveState Win32 Perl in the form of a Perl Module
 - ActivePerl 5.8.6 is included in the standard ProDiscover Distribution (All editions except “for windows edition)
 - All Scripts are executed through the ProDiscover Interface

Function Categories

- General Functions (37)
- Project File Functions (6)
- Data Carving Functions (11)
- Disk and Folder Functions (10)
- File Functions (8)
- Search Result Functions (6)
- Registry Functions (8)
- ACL Functions (3)
- Event Log Functions (7)

Advanced API

- Remote System State Functions (16)
- Search Functions (18)
- Auditing Functions (18)

Installation Notes

- ProDiscover will install Perl upon completion of base installation
- Use default directory for Perl installation
- Choose ProDiscover 'Start' menu option to install Perl module after Perl and ProDiscover base installation is complete

Balance of Power

- The power provided by Perl and ProScripts can be dangerous
 - Perl has extensive “non-write protected” I/O capabilities natively and through the over 500 included modules
 - ProScript also provides the power to crash ProDiscover
- Dangers can be mitigated by testing, experience, and standard forensics methodologies

The 5 Commandments of ProScript

- **WARNING:** While ProScript will not write to evidence disk or images in any way
- **BE ALERT:** the use of Perl File I/O to work with an image file can damage the image.
- **ALWAYS:** Users should ensure all image read functions are performed from ProScript API's.
- **ALWAYS:** Users should understand and test all scripts prior to use on actual evidence images.
- **REMEMBER:** Hardware write blockers can help to protect image files from external applications."

What Makes a Perl script a ProScript?

- Simply adding the directive *use ProScript;*
- And using the ProScript API as needed
- Full Access to the Perl API and other modules is provided

Two ProScript Demos & API Reference Walkthrough

Some Useful Perl Resources

- search.cpan.org/
- aspn.activestate.com/ASPN
- www.activestate.com/support
- www.activestate.com/support/enterprise
- perl.oreilly.com
- www.perl.org
- use.perl.org
- The Gecko book from O'Reilly

Putting Perl to use with ProScript

ProScript Directories

- **\ProScript\Documentation** - contains ProScript API documentation
- **\ProScript\Examples** - contains ProScript API and Perl example scripts
- **\ProScript\Output** – an output directory for script output
- **\ProScript\PerlModules** - contains ProScript Perl module setup files and installation batch file.
- **\ProScript\User Scripts** - contains working ProScripts

Writing and Debugging ProScripts

- The most difficult issue when programming in any language is often identifying syntax errors
 - typing "PSCloseHandel(\$nHandle);" when you meant to type "PSCloseHandle(\$nHandle);"
 - Forgetting a trailing ;
 - Case specific issues such as typing 'Use' or 'use'

Troubleshooting Syntax Issues

- Two techniques to help identify and overcome syntax errors:
- Smart GUI based editors
 - *PrimalScript* from www.primalscript.com/
 - *Komodo* from www.activestate.com
- Using Perl from command line to identify syntax issues

Running ProScripts

- ProScripts must be run from ProDiscover
- Four ways to run ProScripts
 - Button Bar Item
 - Right-Click
 - Tools Menu
 - Command Line
 - ProDiscoverLR.exe project.dft script.pl VarToPass

Testing the ProScript Module

- From ProDiscover main window click on “Run ProScript” on the button-bar
- Choose “browse” and naavagate to “<installation directory>\ProScript\Examples”
- Then choose the script “colors.pl”
- Can be tested without an open project

More Script Walkthroughs

- [React.pl](#)
- [IRAC](#)
- [IRAC2](#)
- [CarvJPGArtifacts.pl](#)

COMPUTER EVIDENCE

Collection & Preservation

- Teaches investigators how to ensure case integrity when dealing with computer evidence
- Provides a practical resource for collection and preservation that will help ensure legal acceptability
- Covers key areas such as rules of evidence, evidence dynamics, network topologies, collecting volatile data, imaging methodologies, and forensics labs and workstations
- Includes a CD-ROM with shareware and commercial demo software tools as well as document templates, worksheets, and references



Networking & Security Series

CHRISTOPHER L.T. BROWN

Thank You
Questions?

Technology
Pathways

703 First Street
Coronado, Ca. 92118

Phone: 888-894-5500
FAX: 619-435-0465

www.TechPathways.com

clbrown@TechPathways.com



Are your
tools
keeping
pace with
the
criminals?