Cisco’s Application eXtension Platform (AXP) Technical Overview

Access Routing Technology Group
View This Presentation in “Slide Show” Mode
Agenda

- ISR Overview
- Branch Architectures
- Application eXTension Platform Overview
- AXP Software Services
- AXP Network and Hardware Services
- AXP Use Cases
- Summary
What Do We Do

We Build
“Integrated Services Routers”
Providing…

Connectivity, Security, IP Telephony, Wireless and Application Solutions

For Enterprise Branch Offices, SP Edges, and Small and Medium Businesses
Traditional Business Solution
Separate Applications and Appliances

- Security
  - Firewall, IDS, and VPN Appliances
- Application Optimization
  - File Engine
- Voice Services
  - Hybrid/Key System
- Data
  - Branch Access Router
- Local Connectivity
  - LAN Switch
Integrated Solution for Advanced Services

Integrated Services Router

Embedded Security
Voice Ready
Application Optimization
L2 Switching
Network Analysis
A Long Cisco History of Innovation...

1985

Pure Data Delivery

Integrated Data, Security, IP Telephony, and Intelligent Application Services

Early Service Convergence

Security, Firewall, VPN

2006

Cisco Access Routers

Integrated Services Routers

Secure, Concurrent Services at Wire Speed
Integrated Services Routers Portfolio

- **3800 Series**: Feature Breadth and Scale at Highest Performance
- **2800 Series**: High Density and Performance for Concurrent Services
- **1800 Series**: Embedded, Advanced Voice, Video, Data, and Security Services
- **800 Series**: Embedded Wireless, Security, and Data

- **SP/Edge**: Head Office
- **Branch Office**: Small Branch
- **SMB**: Small Office, Teleworker, and Mobile
### ISR: Empowered Branch Services

<table>
<thead>
<tr>
<th>Application Intelligence</th>
<th>Mobility</th>
<th>IP Communications</th>
<th>Integrated Security</th>
<th>Foundation Technologies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Improve Effectiveness of Branch Applications</td>
<td>Wireline and Wireless Equivalence—Ubiquitous Secure Connectivity</td>
<td>More Effective Communication and Collaboration Through Application and Infrastructure Integration</td>
<td>Trusted and Protected Business Applications, Legislative Compliance</td>
<td>Integrated, Intelligent Systems</td>
</tr>
</tbody>
</table>

- **Application Intelligence**
  - WAAS module
  - Network analysis module
  - Application-oriented NW module
  - Cisco IOS®-NBAR

- **Mobility**
  - Wireless LAN controller module

- **IP Communications**
  - Cisco Unity® Express module (voicemail, IVR)
  - Cisco IOS: Cisco CallManager Express
  - Cisco IOS: survivable remote site telephony

- **Integrated Security**
  - Intrusion detection module
  - Cisco IOS-VPN
  - Cisco IOS-firewall

- **Foundation Technologies**
  - LAN/WAN connectivity modules
    - Serial, async, DSL, cable, HWICs
    - EtherSwitch® modules
    - Wireless ATM, OC3
Our Strategy Is Integration and Convergence

Overlay Appliances
- Router
- Switch
- Wireless LAN
- WAN/App Optimization
- Security Appliance
- Voice Appliance

Integrated Services Router
- Cisco ISR 3845 with Integrated Voice, Wireless, Video, WAN Optimization, and Switch

Service Interoperability
- Consistency
- Interoperability
- Tested
- High availability

System Support
- Vendor accountability: network partner
- Fewer maintenance contracts

Operational Efficiency
- Fewer devices, management systems, user interfaces
- Simplified troubleshooting

Investment Protection
- Flexibility to evolve through system modularity
Market Acceptance of Integrated Services

- Operational efficiency (70% lower OpEx)
- Services consistency (branch ↔ HQ)
- Guaranteed interoperability
- Investment protection

Overlay Appliances
- 3G Modem
- Router
- Switch
- Wireless LAN
- WAN/App Optimization
- Security Appliance
- Voice Appliance

Integrated Services Router
- Cisco ISR 3845
  - With Voice, Wireless, Video, WAN Optimization, Switch

Millions of Routers Sold
- 5 Months: 36
- 6 Months: 33
- 9 Months: 27
- 18 Months: 18
Market Acceptance for Integrated Services

Four Million Routers Sold

- November 2007
  - 5 Months
  - 38
- 6 Months
  - 33
- 9 Months
  - 27
- 18 Months
  - 18

September 2004

Total Cost of Ownership

- Revenue Loss
- Employee Productivity
- Unplanned Downtime Losses
- Planned Downtime Losses
- Maintenance Contracts
- Facilities (Space, Power, Cooling)
- Implementation Costs
- NMS Costs

Cisco ISR Service Integration:
Up to 70% OpEx Reduction

Service Interoperability
Operational Efficiency
Systems Support
Investment Protection
Branch Architectures

Why, When, Where, and How?
Addressing the New Business Realities of Distributed IT

- Interactive Business
- Collaboration Tools
- Video on Demand
- Wireless Mobility
- Web 2.0

Branches Consume 70% of IT Resources*
Number of Branches is Growing 10% per Year**
WAN Costs Are > 30% of Operational Expenses

Source: *Internet Research Group, 2005; **Nemerts Research Virtual Workplace: Branch Office Strategies
Centralized | Decentralized
---|---
**Pros**
- Simplified administration
- Operational efficiency
- Lower cost

**Cons**
- Performance
- WAN dependence
- Productivity
- Decision making

**Pros**
- Branch performance
- Branch productivity

**Cons**
- Multiple appliances
- Operational inefficiency
- Cost with branch scale
- Administration
Centralized

- Core Business Logic
- Heavy Lifting Computes
- Global Services

Decentralized

- Local Business Logic
- Distributed Computes
- Survivable Services

Pros

- Operational efficiency
- Branch performance
- Branch productivity
- Localized decision making
- Service consistency
- Simplified administration
- Multiple appliances
- Single vendor support
- Operational inefficiency
- Scales hybrid approach
- Cost with branch scale
- Integrated versus “pure play”

Cons

- Administration
- Integrated versus “pure play”
Service-Oriented Network Architecture (SONA) Framework

Collaboration Applications

Collaboration Layer

Middleware and Application Platforms

Cisco Offerings

Application Networking Services

Infrastructure Services

Application Layer

Interactive Services Layer

Networked Infrastructure Layer

New Linkage

Places in the Network

Server

Storage

Clients
AXP Overview

The Architecture Behind Cisco’s New Application Extension Platform
Cisco’s Application eXtension Platform

Host Application Services on Cisco’s ISR

Application Services on Integrated Services Modules

- Network Module (NME): 1.0/1.4 GHz Intel Pentium M; 512 MB to 2 GB RAM; 80 to 160 GB hard disk
- Advanced Integration Module (AIM): Intel Celeron 300 MHz; 256 MB RAM; 1 GB MB Flash
Supported Hardware

- **AIM 102**
  - CPU: 300 Mhz
  - Memory: 256 MB
  - Compact Flash: 1 GB

- **NME 302**
  - CPU: 1.0 Ghz
  - Memory: 512 MB
  - Disk: 80 GB

- **NME 522**
  - CPU: 1.4 Ghz
  - Memory: 2 GB
  - Disk: 160 GB

<table>
<thead>
<tr>
<th></th>
<th>AIM 102</th>
<th>NME 302</th>
<th>NME 522</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco ISR 1841</td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cisco ISR 2801</td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cisco ISR 2811</td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Cisco ISR 2821</td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Cisco ISR 2851</td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Cisco ISR 3825</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Cisco ISR 3845</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>
AXP Services Model

Management Services:
- Application-Level
- AXP-Level
- ISR-Level
- Integrated Services

Security Services:
- SW Trust Chain
- Hardened Linux OS
- Rogue SW Protection
- ACLs
- Stateful FW
- IPS
- Self-Defending NW

Cisco AXP:
- Application eXtension Platform

Software:
- Virtualized hosting environment
- Cisco Linux OS
- Monitor/Configuration API’s

Hardware:
- Standalone CPU, HD, Memory
- NM, AIM service-modules
- ISR 3800, 2800, 1800 support
Redundancy Drives HA

Services Stack

Application Hosting Services:
- > 1 instance per AXP
- > 1 AXP across ISRs
- Both

AXP Blade Services:
- > 1 blade in a single ISR
- > 1 blade across ISRs
- Both

ISR Services:
- > 1 ISR

Network Services:
- Ubiquitous “ilities”
- Integrated Security
- Integrated management

Service Chaining

Virtualization Security Management

“Appliance-Like” Flexible Networking

Multi-Service Integration API “hooks”

Security Performance Mgmt. Availability

Each Layer Affords Service to the Overall Solution, Where Service “Chaining” Comprises Overall Composite Service to Applications
Services Engine Overview

Service Engines Enable **Services Integration** at the Branch While Preserving Router CPU/Memory for Critical Connectivity and Cisco IOS

- **Router integration**
  - Blade control protocol between Cisco IOS and service module

- **Hardware efficiencies**
  - Eliminates separate appliance and preserves rack space
  - Offloads processing/memory to application-specific platform

- **Lower total cost of ownership**
  - Simplifies deployment/maintenance/management

**Service Engine Hardware Offerings**

- AIM: integrated into base routers (1841, 28xx, 38xx)
- NM: slot-based module (28xx, 38xx)
AXP Technical Overview

Dedicated Application Resources
- Dedicated CPU, memory and Disk
- Application separated from core router functionality
- Full networking

Standards-Based Hosting Infrastructure
- Hardened Cisco Linux OS with virtualization
- Complete install/upgrade packaging utilities
- Logging and debugging infrastructure

Programming Support
- Support for Native x86 C/C++
- Java support w/ optional OSGI and Tomcat
- Scripting Support (bash, perl, python)

Value-Added Features
- Serial tunneling providing application access to external devices
- Syslog server to store logs from router and other local devices
- Netflow collector to persist and analyze flows locally

Cisco IOS APIs Integrate the Application into the Network
- Programmatically configure and monitor Cisco IOS
- React to changes in network conditions
- Programmatically Influence Routing, QoS and IP-SLA
- Monitor packets flowing through network
AXP Software Services

Core Software Aspects of the System
Cisco Linux OS Overview

**LSB (Linux Standard Base)**
- Provides a single target for ISVs writing or porting to the Linux platform
  - Asianux 2.0, Debian 4.0, Mandriva Corporate 4.0
  - Red Hat Enterprise Linux 4 and 5
  - SUSE Linux Enterprise 9 and 10
  - Ubuntu 6.06 LTS ("dapper")

**Linux V-Server**
- Creates Virtual instances. Each virtual instance provides a full Linux OS.
- Integrated with AXP Packaging System
- LSB Components of host "Linked" into Guest

**Not Limited to LSB Components**
- LSB Components are installed by default.
- Missing Linux components can be packaged as part of the application.
- Application versions prioritized over LSB components during package install

**Benefits of Environment**
- Consolidation and Separation
- Resource Guarantees
- Independence
AXP Cornerstones and Tenets

- **Predictable** and constant set of resources to third-party applications
- **Discrete** execution environment
- **Extensible** and flexible
- **Integrated** configuration, monitoring, and debugging environment
- **Robust** debugging and troubleshooting facilities
- **Security** against unauthorized software being loaded
AXP Environment

- Third-party applications hosted on ISR blade
- Third-party applications have access to Cisco IOS → advanced applications
  - Modify Cisco IOS configuration
  - Receive notification of events from Cisco IOS
  - Access serial peripherals attached to the ISR
AXP OS
Software Hosting Environment

- Linux-VServer built on top of Cisco’s Linux OS
- Prevents third-party software from interfering with host OS
- Creates virtual instances for application separation
- Managed through host OS via CLI
- Reduced troubleshooting times
- OSGI framework enables remote secure lifecycle management of Java applications
AXP OS
Packaging, Code Signing, and Upgrade

- Packaging mechanism
  Ensures only approved Cisco and third-party applications installable
  Ensures application integrity prior to install

- Installation framework provides third-party upgrades
AXP OS
Native and Interpreted Applications Support

- Embedded Linux environment supports multiple programming languages
  - Java
  - C (native)
  - Perl (interpreted)
  - Python (interpreted)
  - Bash (interpreted)

- Availability to extend existing support via third-party–supplied libraries and interpreters
Virtualization

Using Open Source VServer for Application Sandboxing
AXP
Virtualization

- Multiple third-party applications running simultaneously on a single AXP blade
- Start/stop/control management plane for individual applications
- Complete isolation ensures discrete health state
- Each application runs its own virtual instance
- Security and fault isolation is at the application/process level, not the kernel/OS level
AXP
Linux-VServer

- Container-based virtualization
- Broad-based CPU support, good resource usage, single Linux kernel support (x86 support for AXP—future releases to support additional platforms)
- Efficient use of resources (kernel-level isolation)
  - Process-level security
  - Processes utilize shared resources (no hard partitioning)
AXP
Linux-VServer

Host
/dev /usr /home

Guest VServer 1 (VI1)
/dev /usr /home

Guest VServer 2 (VI2)
/dev /usr /home

Kernel (Shared)
AXP Add-On Packages

- AXP ships preloaded with Cisco OS that supports virtual Instances
- One copy of infrastructure add-ons will be kept in file system (created and signed by Cisco)
- Infrastructure add-ons are instantiated in VI during runtime startup (shared files)
- Third-party add-ons are loaded by third-party vendor

Shared Files Results in Increased Efficiency in System Resource Utilization
AXP
Installation Process

1. Determine dependencies based on third-party application’s AXP manifest file
2. Create VI and register third-party application in AXP CLI
3. Implement shared files by creating hard links
4. Install third-party application in VI (may overwrite hard links from Step 3)
5. Execute post-install configuration script (resource allocation, network configuration, etc.)
CLI Capabilities

Command Line Interface
AXP CLI Structure

CLI Extensions (Third-Party CLI Integration)

- Custom CLI service
  
  CLI server plug-in tools
  
  Tools used at third-party development time to validate, process and package the custom CLIs

  CLI distribution service
  
  Interface between AXP CLI server and third-party applications; handles distribution of custom CLIs from AXP CLI server to third-party application that CLIs belong to

- Custom CLI definitions
  
  Cisco-defined syntax

- Custom CLI actions
  
  Java, C, Shell script implemented via Cisco-defined signatures
Packaging Applications in AXP

Packaging Is a Core Element of AXP’s Value Proposition
Development Process—Production

- **SDK**
- **Develop**
- **Package**
- **Installation**

**Development Machine**
- Linux (FC4)
- Tools (compiler, make, IDE)

**SDK** (packaging scripts, API headers/libs for various languages)

- **Source Code**
- **Compile**
- **Binaries**
- **Package**
- **AXP .pkg**

**Auth File + Private Key**

**ISR AXP Blade**

- CLI> Software Install
- Add URL
- ftp://.....AXP.pkg
Application Packaging

Vendor’s X.509 Certificate (Includes Public Keys) + Checksum of Certificate Encrypted with AXP Private Key

Vendor’s Private Key

Application Files

Sign Application Files

Compress Application Files

Signed Application Bundle

CLI Software Updates
Packaging and Bundling Applications
Issuing Application Development Authorization

1. Generate certificate request (private/public keys)
2. Request certificate signing
3. Respond with signed X.509 certificate
4. Request software development authorization (include signed certificate)
5. Verify certificate authority
6. Respond with software development authorization
Application Monitoring, Logging, and Tracing

Monitoring, Logging, and Tracing of Applications in VIs
# Application Status

## Monitoring State and Health

AXP provides an API call for an application to report its health.

Note: AXP will not actively monitor application health values; the application has to report its health to the AXP host.

<table>
<thead>
<tr>
<th>Application</th>
<th>State (VI)</th>
<th>Health (App)</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>App_A</td>
<td>Online</td>
<td>Initializing</td>
<td>VI Is Running; App Is Starting</td>
</tr>
<tr>
<td>App_B</td>
<td>Online</td>
<td>Alive</td>
<td>VI Is Running; App Is Starting</td>
</tr>
<tr>
<td>App_B</td>
<td>Online</td>
<td>Down</td>
<td>VI Is Running; App Is Down</td>
</tr>
<tr>
<td>App_C</td>
<td>Online</td>
<td>Down</td>
<td>VI Is Down</td>
</tr>
</tbody>
</table>
## Application Logging
### Logging Levels

<table>
<thead>
<tr>
<th>Syslog Level</th>
<th>Meaning</th>
<th>CLI Access?</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOG_EMERG</td>
<td>System Is Unavailable</td>
<td>No</td>
</tr>
<tr>
<td>LOG_ALERT</td>
<td>Action Must Be Taken Immediately</td>
<td>No</td>
</tr>
<tr>
<td>LOG_CRIT</td>
<td>Critical Conditions</td>
<td>No</td>
</tr>
<tr>
<td>LOG_ERR</td>
<td>Error Conditions</td>
<td>Yes</td>
</tr>
<tr>
<td>LOG_WARNING</td>
<td>Warning Conditions</td>
<td>Yes</td>
</tr>
<tr>
<td>LOG_NOTICE</td>
<td>Normal, but Significant, Condition</td>
<td>Yes</td>
</tr>
<tr>
<td>LOG_INFO</td>
<td>Informational Message</td>
<td>Yes</td>
</tr>
<tr>
<td>LOG_DEBUG</td>
<td>Debug-Level Message</td>
<td>No</td>
</tr>
</tbody>
</table>

All log levels are available to the application, however, CLI will reduce configuration to four log levels shown above.
Application Tracing

- Support for both system level and application level tracing support
  - Third-party software can output internal trace information
  - Both third-party app and OS level logging will be logged to syslog

- Tracing most useful for debugging system-level defects

- Application level tracing within the virtual application environment can be merged with logging (syslog) with no loss of its functionality
Application Logging
Logging Correlation Between AXP and Applications

- Log messages are correlated with each other between third-party applications and AXP components
AXP

Syslog Server

- Optional enabling of a syslog server on AXP OS
  - Allows AXP device to collect log messages from other physical and virtual devices (e.g., virtual instances)
  - Syslog server will accept messages from any source on the network, no authentication support will be considered
  - Maximum size of log file configurable by customers

- Disk requirements
  - Syslog feature consumes available storage space (shared with virtual instances), within limits set by administrator
  - Proper planning required

- Log files available for download
  - Via CLI command
  - Linked to virtual instances (/var/log directories)
Resource Controls

Controlling Consumption of System Resources in Application Hosting Environments
AXP
Application Resource Utilization Control

Goal

Create a Predictable Application Hosting Environment That Does Not Require Complex Configuration

- Resource limits defined by application developer in resource file in package
  Reside in application’s manifest file
  (e.g., Mem = xMB, Disk = yMB)

- Resource limits based on platform capacities
  Example: platform CPU index = 10000
  Example: setting application CPU usage index to 6000 means 60% of CPU resources will be utilized (maximum)
Resource Management

- AXP automatically manages resource utilization globally, across VServer instances
- Example: “show resource limits”

<table>
<thead>
<tr>
<th>Application</th>
<th>CPU (Index)</th>
<th>Memory (MB)</th>
<th>Disk (MB)</th>
<th>Log (MB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>App_#1</td>
<td>1500</td>
<td>160</td>
<td>150</td>
<td>5</td>
</tr>
<tr>
<td>App_#2</td>
<td>5000</td>
<td>256</td>
<td>15,000</td>
<td>5</td>
</tr>
<tr>
<td>System Total Allocated</td>
<td>6800</td>
<td>496</td>
<td>15,330</td>
<td>–</td>
</tr>
<tr>
<td>Resources Available</td>
<td>3200</td>
<td>4</td>
<td>60,985</td>
<td>–</td>
</tr>
</tbody>
</table>
Enforcing Resource Constraints

- During application install the following checks will be performed:
  
  Application resource profile is compared to available resources to determine if resources available
  
  Installer will update global resource tracking database when new applications are installed
AXP SDK

Third-Party Development Environment
AXP SDK

- Cisco will provide 3rd-party vendors with an SDK for the Linux environment; the SDK will include:
  - The standard infrastructure add-on packages
  - Information about recommended GCC compiler
  - AXP Guest OS libraries and header files
  - AXP server libraries and any associated header files
  - Tool to simplify manifest creation, including specification of dependencies
  - Toolkit for developing custom application CLI
  - Application packaging and signing tool
Debugging Applications

Third-Party Development Debugging Capabilities
Debugging

- Remote debugging capable
  - Remote debuggers written in C/C++ only
  - Remote Java debugging requires application developer to include remote debugger (e.g., Jdb) or compile remote debug stubs into their application
  - Debugging packages removed for production

- SSL Tunneling
  - OpenSSH
  - Enables debugging more easily through firewalls

- SSL Tunnel Authentication
  - Public key or;
  - Password authentication based

- Disabled in production blade by default
Network and Hardware Services

Programmatic Interface to Integrate with Network Services
AXP Network Support

Internal Interface Connects Blade to the Router

- Integrated-Service-Engine X/0 is the interface on Cisco IOS
- Eth0 is the interface on the Linux side

Virtual Interfaces “Bind” to Interface

- Flexible use of available interfaces
- Sub-interface support
- 802.1Q (VLAN) Encapsulation support
- VRF support

External (NME only) Interface

- Exposed to Linux as Eth1
- Virtual instances optionally bind to interface
- Externally routable interface
- Configurable as default route

Flexibility Means Usability:

- Migrating existing applications from servers to AXP made easy
- Multiple applications benefit from single or multiple subnets
- Security provisions at blade-level or application-level
AXP
Cisco IOS APIs

- Cisco IOS Configuration API
  Allows the application to programmatically change the configuration of the router
  Utilized for real-time changes “on the fly”

- Cisco IOS Information API
  Provides query capabilities into Cisco IOS; data contained in show commands, SNMP and exec commands can be retrieved through this API

- Cisco IOS Trigger API
  Facilities to send notifications to the application based on events in Cisco IOS; allows the application to react to network conditions, changes to Cisco IOS configuration and other Cisco IOS events

- Packet API
  Mechanisms to send packets to third-party application for analysis or processing
AXP
Cisco IOS APIs

Application

Config_command(s)
Exec_command(s)

Return_value(s)

Cisco IOS Service API

Cisco IOS CLI

Cisco IOS Services
AXP EEM API

- Event Detectors
  “Watch for events of interest”

- EEM Server
  The “brains” of the system

- Policies (scripts)
  Applets
  TCI-based

* All within Cisco IOS
Cisco IOS Event Notification

EEM—Embedded Event Manager

- Embedded Cisco IOS events trigger an XML-RPC call to an application (e.g., config change, interface state change)
- Applications use API’s to listen for events that they are interested in hearing about
- Configured via the CLI
**AXP**

**L2/L3 Networking Considerations**

- **Individual Virtual Instances**
  - Networking configurations decoupled from virtual instances
  - Virtual instance may bind to more than one network interface at a time

- **Interface support**
  - Physical Ethernet sub-interfaces
  - VRF support
  - VLAN interfaces (802.1Q)
  - SSH

- **ISR configurations**
  - Standard ISR IP routing configuration commands used
  - ISR IP configuration context separate from AXP IP configuration context
External Device Support

Applications Have the Ability to Connect to Devices External to the Blade
External Devices
Virtual TTY Devices

Cisco IOS Host Serial Port

- Local AXP host TTY device interacts with the external Cisco IOS serial device
- Device name: /dev/router.aux
- Reverse Telnet session established
- Serial port settings (e.g. baud rate) configured through Cisco IOS
Packet Services

Using Cisco IOS Packet Services to Forward Packets to the AXP Service Module
Application Access Modes

Direct Access—Client sends traffic directly to application IP address (standard server model)

Intercept—Traffic is sent to a remote host and is intercepted by the router and delivered to the application

Promiscuous—A copy of each packet is sent to the module for monitoring/analysis. Flow of packet is unaffected
AXP—Direct Access/Intercept Mode

**Enablers:**
- IP Routing and PBR (Optional)
- HSRP

1) Packet enters ISR (HSRP interface)
2) AXP subnet is entry in FIB
3) Packet received on Vserver instance “service” IP interface
4) Application consumes packet
5) Application has option to originate packets and send over external GE interface or internal GE interface (ISR)

Client

ISR

GE

VI_1

AXP
AXP—Promiscuous Mode
Sniffer Type Functionality (Like SPAN)

Enablers:
- NAM, Packet Capture, or
- RITE

1. Packet enters ISR (HSRP interface)
2. CEF mechanism (e.g., NAM P.C.) spawns second identical packet
3. One packet continues through CEF stack
4. One packet sent to Vserver instance “service” IP address
HA Problem Statement

Fundamental Challenges in Achieving High Availability in General
Redundancy Drives HA

Redundancy Stack

Application Redundancy:
- > 1 instance in ISR
- > 1 instance across ISRs
- Both

AXP Blade Redundancy:
- > 1 blade in a single ISR
- > 1 blade across ISRs
- Both

ISR Redundancy:
- > 1 ISR

Clients Redundancy:
- Dual-homed or -interfaced (optional)

Dynamic Bypass

AXP

VI_1

ISR

Client Pool_A

Each Layer Needs to Be Considered Against the Following Criteria:
1. Active-Active Load-Sharing Load-Balancing
2. Active-Standby

It Is Critical to Impose System-Wide HA Mechanisms
Application Redundancy
Single Application, Single Blade, Single or Redundant ISRs

**Redundancy Stack**

**Application Redundancy:**
- PBR-based on “next-hop” config in Cisco IOS
- Dynamic bypass if VI failure

**AXP Blade Redundancy:**
- N/A (see multiple blade example)

**ISR Redundancy:**
- Multiple IP sub-interfaces (optional)
- Master HSRP (if 2nd ISR)

**Clients:**
- Dual-homed or -interfaced (optional)
- DG = HSRP VIP on ISR

**Dynamic Bypass**

- If application or blade is down, PBR tracking if set to down → triggers PBR mechanism fallback to destination-based routing (traditional)

**Forwarding path from VIs is either over GE external or GE internal (back to ISR)**

[Diagram showing the flow of data from Client Pool_A and Client Pool_B to VIs, AXP, and ISR_x with PBR mechanisms and redundant paths labeled.]
Application Redundancy
Single Application, Multiple Blade, Redundant ISRs

Redundancy Stack

Application Redundancy:
- PBR-based on “next-hop” config in Cisco IOS
- “Load-sharing” via inversion

AXP Blade Redundancy:
- Physical + host
- If blade fails, then PBR keepalive to VI triggers flip

ISR Redundancy:
- Multiple IP sub-interfaces (optional)
- Master HSRP (if 2nd ISR)

Clients:
- Dual-homed or -interfaced (optional)
- DG = HSRP VIP on ISR

Load-Sharing VIs

- If application or blade is down, PBR tracking if set to down → triggers PBR mechanism fallback to destination-based routing (traditional) for that instance only
- Surviving application must be able to handle all traffic when failover occurs

Forwarding path from VIs is either over GE external or GE internal (back to ISR)
Application Redundancy
Multiple Application, Multiple Blade, Redundant ISRs

**Redundancy Stack**

**Application Redundancy:**
- PBR-based on “next-hop”
- “Load-sharing” via inversion

**AXP Blade Redundancy:**
- Physical + host
- If blade fails, then PBR keepalive to VI triggers flip

**ISR Redundancy:**
- Multiple IP sub-interfaces
- Master and Standby HSRP

**Clients:**
- Dual-homed or -interfaced
- DG = HSRP VIP

**Load-Sharing**

- AXP
- PBR
- Client Pool_A
- Client Pool_B
- To ISR_y

- AXP
- VI_1
- VI_2
- ISR_x

- AXP
- VI_1
- VI_2
- ISR_x
Application Redundancy
Multiple Application, Single Blade, Redundant ISRs

Redundancy Stack

Application Redundancy:
- PBR-based on “next-hop”
- “Load-sharing” via inversion

AXP Blade Redundancy:
- N/A (see multiple blade config to extrapolate to this scenario)

ISR Redundancy:
- Multiple IP sub-interfaces
- Master and Standby HSRP

Clients:
- Dual-homed or -interfaced
- DG = HSRP VIP

Load-Sharing VIs/Active-Standby ISRs

- PBR
- ISRs
- AXP
- Clients Pool_A
- Clients Pool_B
Application Redundancy
Multiple Application, Single Blade, Redundant ISRs

Redundancy Stack

**Application Redundancy:**
- PBR-based on “next-hop”
- “Load-sharing” via inversion

**AXP Blade Redundancy:**
- N/A (see multiple blade config to extrapolate to this scenario)

**ISR Redundancy:**
- Multiple IP sub-interfaces
- Master and standby HSRP

**Clients:**
- Dual-homed or -interfaced
  - DG = HSRP VIP (Pool masters on different ISRs)

Load-Sharing VIs/Active-Standby ISRs

- AXP
- VI_1
- VI_2
- PBR
- ISR_x
- Client Pool_A

- AXP
- VI_1
- VI_2
- PBR
- ISR_x
- Client Pool_B

Diagram showing the connections between the ISRs, PBRs, and various VIPs.
AXP Security
Layers of Protection

Cisco IOS Security Application Container

- Linux Security
- Cisco IOS Security
- Network Security

Cisco IOS

Security Application Container

- Application

- Hardened Linux OS
- End-to-end SW Trust
- Intrusion Prevention
- ACL access control
- L2TP Client initiated Tunneling
- IPSEC w/ DMVPN
- Self-Defending Network
- Rogue Software Protection
- End-to-end SW Trust
- Intrusion Prevention
- ACL access control
- L2TP Client initiated Tunneling
- IPSEC w/ DMVPN
- Self-Defending Network
- Rogue Software Protection
Packaging and Bundling Applications
Trust Chain Mechanism for Application Vendors

- Unauthorized software will not be allowed to be loaded into operating system
  - Enforced through cryptographic signatures
  - Verify packages have been signed by Cisco

- Application vendors will not have access to private Cisco keys for package signing
  - Need to manage their own public/private key pairs

- Managing permissions
  - Cisco responsible for managing permissions to install software into AXP environment
  - Cisco will provide third-party vendors a checksum of their X.509 certificate encrypted with AXP OS private key → key becomes authorization
AXP Management
Use Case #1: Install/Update

Network Administrator (IT Operations)

1. Open Change Request Ticket
2. Install ISR + Services Modules including AXP Module
3. Close Change Request Ticket

Repeat x 1000

1. Open Change Request Ticket
2. Install Application
3. Close Change Request Ticket

Virtualized OS
Java Application
OSGI
Perl/Python
Java
Virtualized OS
Extensible Cisco IOS-like CLI
Cisco Linux OS

Cisco IOS Interface

Cisco IOS

AXP Module

Cisco ISR

LAN, WAN, Local

LAN or WAN
Use Case #2: Administer/Configure

1. Open Change Request Ticket
2. Administer
   • SSH to ISR, use CLI
   • Change routing, re-IP
   • Run sniffer
   • Run bulk-installation script
3. Close Change Request Ticket

Repeat x 1000

Network Administrator
(IT Operations)

Application Administrator
(IT Operations)

1. Open Change Request Ticket
2. Administer
   • Patch application
   • Install open-source utilities
   • Collect app log files
   • Start/stop application
   • Issue OS commands
   • Run bulk-admin script
   • Troubleshoot
3. Close Change Request Ticket
Use Case #3: Monitor/Notify

1. Log, Open Ticket
2. Assess
3. Notify/Escalate

SNMP / SMTP Alert

Monitoring Tools
- BMC Patrol
- NetIQ Security Manager
- Microsoft Operations Manager
- Micromuse NetCool

Enterprise Systems/Network Mgmt Tool
- IBM Tivoli
- HP Openview
- CA Unicenter

Network Administrator (IT Operations)

Application Administrator (IT Operations)

24x7 Network/Security Operations Center (IT Operations)
High-Level Architecture

Cisco ISR

Cisco IOS

AXP Module

Management Agent

Plug-In

Plug-In

Plug-In

Plug-In

API

Application

Middleware

System (AXP OS)

Network (Cisco IOS)

Cisco IOS Interface

Management Console

API

Alerts

Management Server

Open Schema

Database

24x7 Network/Security Operations Center

Reports and Analysis

Scripts, Workflow Processes, External Systems

Scripts, Workflow Processes, External Systems

API

API

API
# AXP Management Value Proposition

<table>
<thead>
<tr>
<th>Feature</th>
<th>Benefit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Centralized, Proactive Monitoring</strong></td>
<td>▪ Integration with existing, standard IT processes</td>
</tr>
<tr>
<td>▪ Forwarding of AXP application and ISR</td>
<td>▪ Implement high application SLAs</td>
</tr>
<tr>
<td>alerts to central NOC/SOC management tool</td>
<td></td>
</tr>
<tr>
<td><strong>Centralized, Automated Administration</strong></td>
<td>▪ Quicker, less error-prone administration of large number of applications and ISRs</td>
</tr>
<tr>
<td>▪ Bulk administration of both AXP</td>
<td>▪ Lower TCO/operating costs</td>
</tr>
<tr>
<td>applications and ISR</td>
<td>▪ Service provider competitive advantage =&gt; increased uptime/reliability and quicker, more reliable rollout of new services</td>
</tr>
<tr>
<td>▪ API and automation support</td>
<td></td>
</tr>
<tr>
<td><strong>Out-of-Box, Integrated Stack Management</strong></td>
<td>▪ More complete information about application and dependencies</td>
</tr>
<tr>
<td>▪ Network (network traffic, routing, SNMP)</td>
<td>▪ Better decisions with regards to service outages</td>
</tr>
<tr>
<td>▪ OS / System (CPU, RAM, Disk, syslog)</td>
<td>▪ Partner/Customer focuses on core business i.e. application development, not management (shorter time-to-market / development cycle)</td>
</tr>
<tr>
<td>▪ Middleware (web/app server, database)</td>
<td>▪ Consolidated, centralized, holistic view of business application/service</td>
</tr>
<tr>
<td>▪ Application logfiles, process up/down, process control</td>
<td>▪ Decreased problem identification/resolution resulting in increased application uptime</td>
</tr>
<tr>
<td><strong>Easy-to-use Application Management APIs</strong></td>
<td>▪ Quickly implement rich application management functionality</td>
</tr>
<tr>
<td>▪ Custom application management and monitoring</td>
<td>▪ Monitor business application health and events</td>
</tr>
<tr>
<td><strong>Seamless Integration in Common Interfaces</strong></td>
<td>▪ Implement/improve business application SLAs</td>
</tr>
<tr>
<td>▪ GUI / CLI / API</td>
<td>▪ Minimize training costs with consistent administration interface</td>
</tr>
<tr>
<td></td>
<td>▪ Maintain native / legacy tool interfaces</td>
</tr>
</tbody>
</table>
AXP Use Cases

Design/Configuration of Common Deployment Models
Distributed Application Deployments
Benefit from Network + Application Integration

- Consolidation with network devices reduces operational complexity
- Network integration enhances security
- Intelligent management for WAN outage
- Enhanced user experience
AXP Use Cases—in a Nutshell

- Augments the functionality of the router with value-add Cisco supported services, open source components, third-party applications and homegrown utilities

**AXP Central Management**
- Software Management (install, upgrade, patches)
- Application/platform Configuration + monitoring
- Extensible architecture to manage custom apps

**Network Services**
- AAA Server
- DNS Server
- NTP Services
- File Services
- Syslog Server

**Home-Grown Utilities**
- Management Agents
- Monitoring tools
- Custom scripts
- Netflow Analysis

**Applications**
- Business Apps
- Vertical Apps
- Telephony apps
- Software Mgmt Systems
Network-enabled Thin Branch Applications

- Edge clients utilize centralized application in the DC
- If the central instance is not available, the router intercepts and forwards the request to local **AXP-based thin version of application** for survivability
  
  Application listens to WAN or central application failure and switches over

- Application resumes normal operation when central version is up
  
  Syncs up information with AXP application

![Diagram of Network-enabled Thin Branch Applications]

Utilizes Network Awareness for Application Survivability
Network-Aware Applications

Use Case: Router configured with high-bandwidth link for primary, low-bandwidth link for failover. Application utilizes high-bandwidth link to provide services to local clients.

Network Failover with Typical Application
- Router fails over to low-bandwidth link
- Application is not aware of the drop in capacity
- Continues with normal operation
- Loss of service and unpredictable behavior occurs.

AXP Network-Aware Application
- Router notifies application of bandwidth change
- Application queries router to determine the current state of the outage
  - Dynamically alters router settings based on business rules
    - Settings based on business situation (time of day)...
- Application alters behavior based on new information
  - Requests low-bandwidth version of data
  - Limits requests to high priority items only
High Availability
Network-Based Failover for Edge Applications

- Edge clients utilize a local AXP based embedded application
  - Client initiates a request
  - Router intercepts request and delivers to the AXP based application
  - AXP application responds to client

- If the local instance fails, the router bypasses the intercept and forwards the request to a centralized global failover server

- Utilizes the network for redundancy
- Supports Standard TCP-based edge applications
- Saves on Costly HA infrastructure in the edge
Use Case: Custom Network Services

**Problem**
- IT policy prevents deployment of additional physical servers in the branch
- Difficult to perform basic network management or troubleshooting

**Solution**
- Custom network utilities to monitor and troubleshoot the network
- X-Windows access allows admin to view packets and utility results in real-time
- Custom SLA utilities to measure performance
- Augmented with Cisco-supported network utilities

**Benefits**
- Platform to enable service providers to enable their own tools and monitoring utilities
- Management of customer networks—new services
- Local survivability of business services
Use Case: Packaged Network Utilities

Problem

- Various core network-based services such as DNS, DHCP, and AAA need to be resident in each remote site but centrally managed

Solution

- ISR with AXP hosting multiple common network utilities (DNS, DHCP, TFTP, AAA)

Benefits

- Better service to end-customer (performance, availability)
- Integrated solution with lower TCO than other solutions
- No additional appliances; conservation of physical space
- Centrally managed

Local Network-Based Utilities
For LAN-Side Clients, WAN Outage Survivability, etc.

- AAA
- DNS
- DHCP
- NTP
- ...
Use Case: UC Apps

Examples:

- **Speech capabilities at enterprise branches**
  
  Speech capability on ISR blade can be leveraged by multiple applications: IVR, Cisco Unity Express, Contact Center Express

- **Branch voice recording**
  
  Light weighted recording/retrieval modules on ISR to support ad-hoc recording, to minimize WAN bandwidth and support recording survivability

• UC apps. Ported on AXP
• UC apps inter-work with CME
• AXP offers hosting environment & Cisco IOS integration
Case Studies

Customer Examples of Application Hosting Environments
Use Case: Lean and Rich Retail Store

Problem
- Retailers want to centralize applications for smaller stores in the DC
- Business continuity is a challenge when WAN does down or becomes slow

Solution
- Thin version of POS & other retail applications on AXP blade on ISR talking to centralized mothership
- Applications interacts with AXP network APIs to go in survivable mode when WAN deteriorates

Benefits
- Lowers infrastructure costs while maintaining business continuity
Use Case: Time Tracking/Workflow

Workforce Management (Employee Scheduling and Optimization); Front-End Is Employee Clock-In/Clock-Out

**Problem**

- Capturing and consolidating time clock punches critical for the bottom-line in low-margin business
- Need way to address other unique services without high cost/overhead to retail store

**Current Solution**

- Clock-in/clock-out via dedicated hardware punch clock
- Store-side clock-server app runs on in-store server and synchronizes up with central app

**New Solution**

- Clock-in/clock-out via Cisco IP phones with other value added services

**Benefits**

- Reduced infrastructure with store-side clock-server app running on AXP
- Intelligent buffering and sync of data when WAN degrades
Ex: Integrated Environment (Logical)

IP Telephony Endpoint

Clock Server

Media Server

PCI

UC4R

Intelligent Retail Network

Store

Data Center

Application Server

PCI

UC4R

AxP

CUAE

UCAD

Browser

System Control Panel

Media Server

PC

CUAE

Presentation_ID

100

© 2008 Cisco Systems, Inc. All rights reserved. Cisco Confidential
Embedded Network Services = Leverage Network "ilities"
Managed Service Providers
Use Case #2

MSP #1 w/Custom Developed CPE Device

- Access router built with Open Source Technologies
- Integrated value-added proprietary services
  - Custom Remote management and monitoring services
  - Dynamic IP-SLA/QoS services
  - Niche Security services
  - CDN, News feed support, Credit Card support …
- AXP enables MSP to utilize Cisco routers to solve their business needs
- 10,000 non-Cisco end points

Managed Service Provider #2

- Looking for a way to differentiate themselves from competition and add additional revenue
- Remote management, monitoring of customer networks.
- Time of the day routing
- Distributed control domains
- Security Services
- Proprietary services
Use Case: Dynamic Networking Services

Problem

- Wartime mobile command centers require reliable and adaptive communications network
- Need a responsive network to dynamically change routing, servers used for messaging/alerting, upstream points of contact, encryption keys
- Hardware/software footprint must be reliable, simple, and integrated

Solution

- ISR with AXP blade
- Routing and policy decisions based on location and situation
- Dynamic, fine-grained control of router
- Direct connectivity to proprietary serial devices (e.g. mobile radio units) using ISR serial port

Benefits

- Integrated interfaces to configure and monitor Cisco IOS
- Cisco IOS event triggers allow application to dynamically react to changes in the network
- No extra server to support at the edge
- Central, integrated device management
**Use Case: Connected Healthcare**

**Problem**
- Doctors struggle to care for patients without knowledge of past treatments / illnesses
- Dangerous medical mistakes, wrong prescriptions

**Solution**
- Healthcare Connector Application
- Cisco ISR 1841 w/ integrated AXP AIM blade
- USB support for card readers other devices
- Programmatic control of VPN by application

**Benefits**
- Meets stringent privacy and encryption standards for health record transmission
- Fully-integrated solution (HW/SW platform) with utilization of ISR USB ports for integration of smart card readers
- Easily managed for physician’s office and health clinics
- Low-cost

![Diagram](Image)
Summary
AXP Technology Value Proposition

- **Opex reduction with physical consolidation**
  - One-box solution
  - Consolidated maintenance

- **Manageability and operations**
  - Centralized deployment and upgrades
  - Monitoring

- **Hardened “appliance” like characteristics**
  - Network device like reliability
  - Easy on-boarding

- **Security**
  - Secured through router
  - Locked down OS

- **Future proofing**
  - Flexible SW upgrades to add new applications

- **New application capabilities enabled through network integration**
  - Intelligent HA/failover
  - Network aware application behavior