IEEE 1711-2010

Industrial Control Systems Security

A Perspective on Product Design

Tien Van tvan@sequi.com
Tracy Amaio, Ph.D. teamaio@sequi.com

INDUSTRIAL CONTROL SYSTEMS (ICS)

Characteristics:

- Long operational life (10+ yrs)
- Small to large geographic area
- Highly complex and found everywhere
- Field RTUs/PLCs are in the open, most are unprotected
- Routable (TCP/IP) protocols
- "Legacy" non-routable (serial) protocols:
  - Radio, leased line, dial-up, and multi-drop links
  - Low data throughput
  - Slow telemetry polling
  - Modbus, DNP3 protocols (MTU – RTU communications)
  - Difficult to add security to existing software
  - Little/no auditing, logging

VULNERABILITIES – THE PROTOCOLS

Most legacy protocols do not have authentication making them easy to exploit and attack

- Modbus was designed to program controllers by sending Read and Write I/O registers commands, for example:
  - ✓ List defined points and their values
  - ✓ Request information about Modbus servers, PLC configurations…
  - ✓ Clear, erase, or reset diagnostic information
  - ✓ Force slave devices into "listen only" mode

Modbus RTU	Start 3.5 char time	Addr 1-byte	Func 1-byte	Data 0-252	CRC 2-byte	End 3.5 char time
Modbus ASCII	Start :	Addr 2-byte	Func 2-byte	Data 2x (0-252)	LRC 2-byte	End CR LF

- DNP3 has source and destination addresses that can be useful in Man-in-the-Middle attacks, such as:
  - ✓ Turn off unsolicited reporting to stifle specific alarms
  - ✓ Spoof unsolicited responses to the Master to falsify events and trick the operator into taking inappropriate actions
  - ✓ Issue unauthorized stops, restarts, or other functions that could disrupt specific operations

DNP3 [Header + Data] Max frame size: 292 bytes							
Header =	0x05 1-byte	0x64 1-byte	Len 1-byte	Ctrl 1-byte	Dst 2-byte	Src 2-byte	CRC 2-byte

COMMON ICS ATTACKS

- **Maintenance port**

  To install a malicious program

- **Spoofing**

  To masquerade as another to initiate an unauthorized action

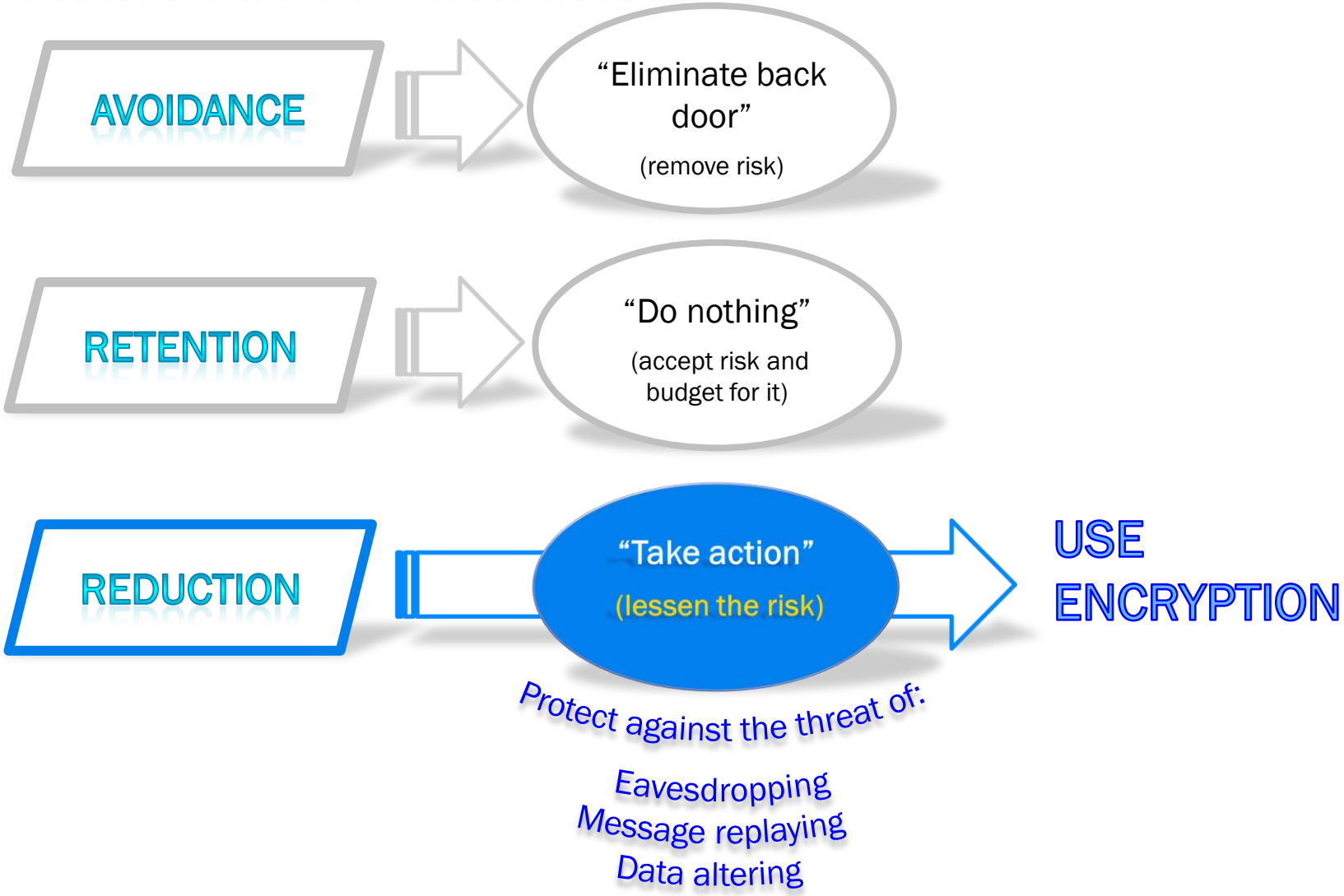
- **Replay**

  To record and retransmit valid data (manipulating time variable) to trigger unpredictable results

- **Man-in-the-Middle (MITM)**

  To intercept, alter, and relay a communication message
  - A simple radio MITM can be setup by a combination of directional transmitter & jammer

RISK MANAGEMENT - LEGACY SYSTEMS



LEGACY RETROFIT ISSUES

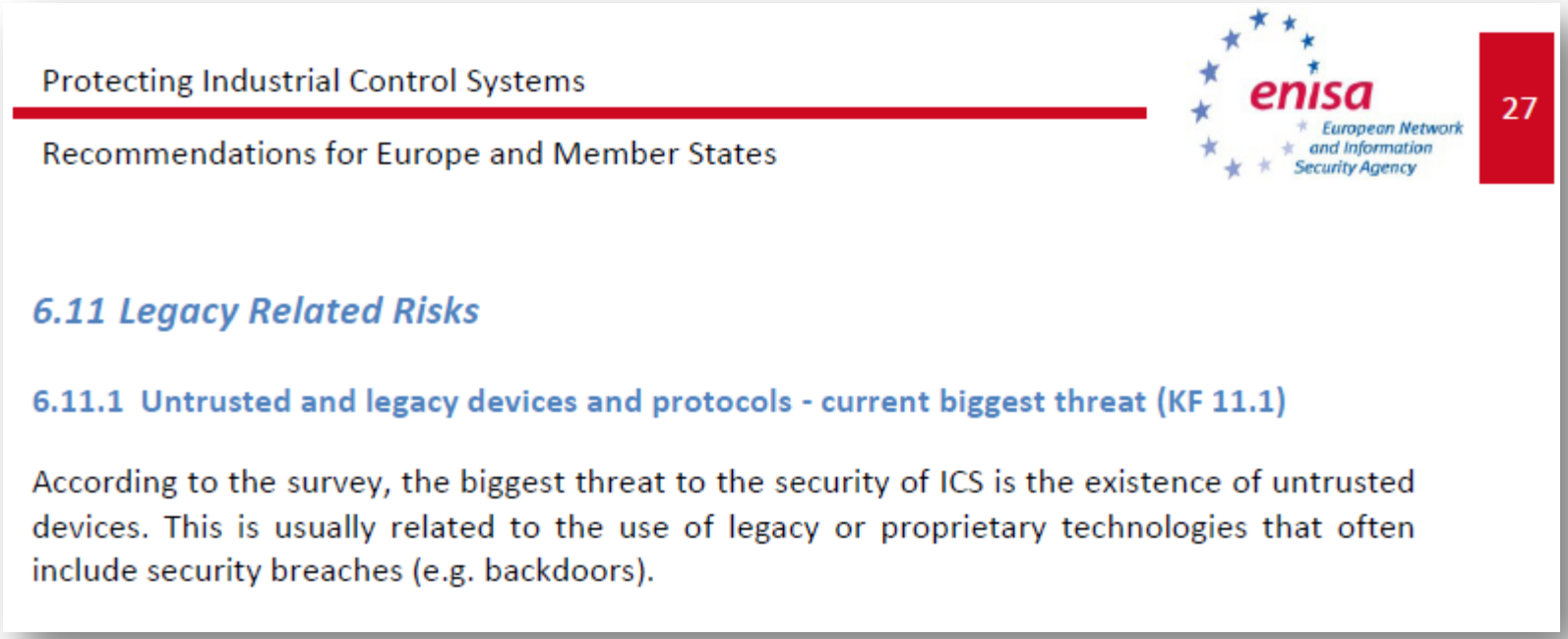
Top Questions	What Customers Want
How do we plan for migration?	• To choose when & how much security to apply • To have encrypted & clear communications on the same channel …Easy migration
Do we make changes to our ICS software or equipment?	Not to change existing ICS software or equipment …Easy installation
Do we make changes to our operational control?	Not to change existing operational control …Hassle-free
Will it impact performance?	Strong security without impacting performance …No impact
Will it impact our existing configuration?	Not to change existing configuration …Flexibility *Support for:* • *Modbus RTU/ASCII, DNP3* • *Async 300 to 115200 bps* • *Point-to-point and multi-drop* • *Radio, dial-up, leased lines*

ICSJWG 2011 FALL CONFERENCE (LONG BEACH CA)

- SEQUI presented “IEEE 1711-2010 Security for Legacy SCADA Protocols”
  http://www.us-cert.gov/control_systems/icsjwg/presentations/fall2011/D1-09-0200pm_Track2_Amaio-Van_rr_Title-IEEE1711-2010SecforLegSCADAProt.pdf

CURRENT BIGGEST THREAT TO ICS

- European Network and Information Security Agency (ENISA) conducted a survey to identify threats, risks, and challenges to ICS and found that untrusted and legacy devices and protocols are the biggest threat to security of ICS.



Protecting Industrial Control Systems

Recommendations for Europe and Member States

enisa European Network and Information Security Agency

27

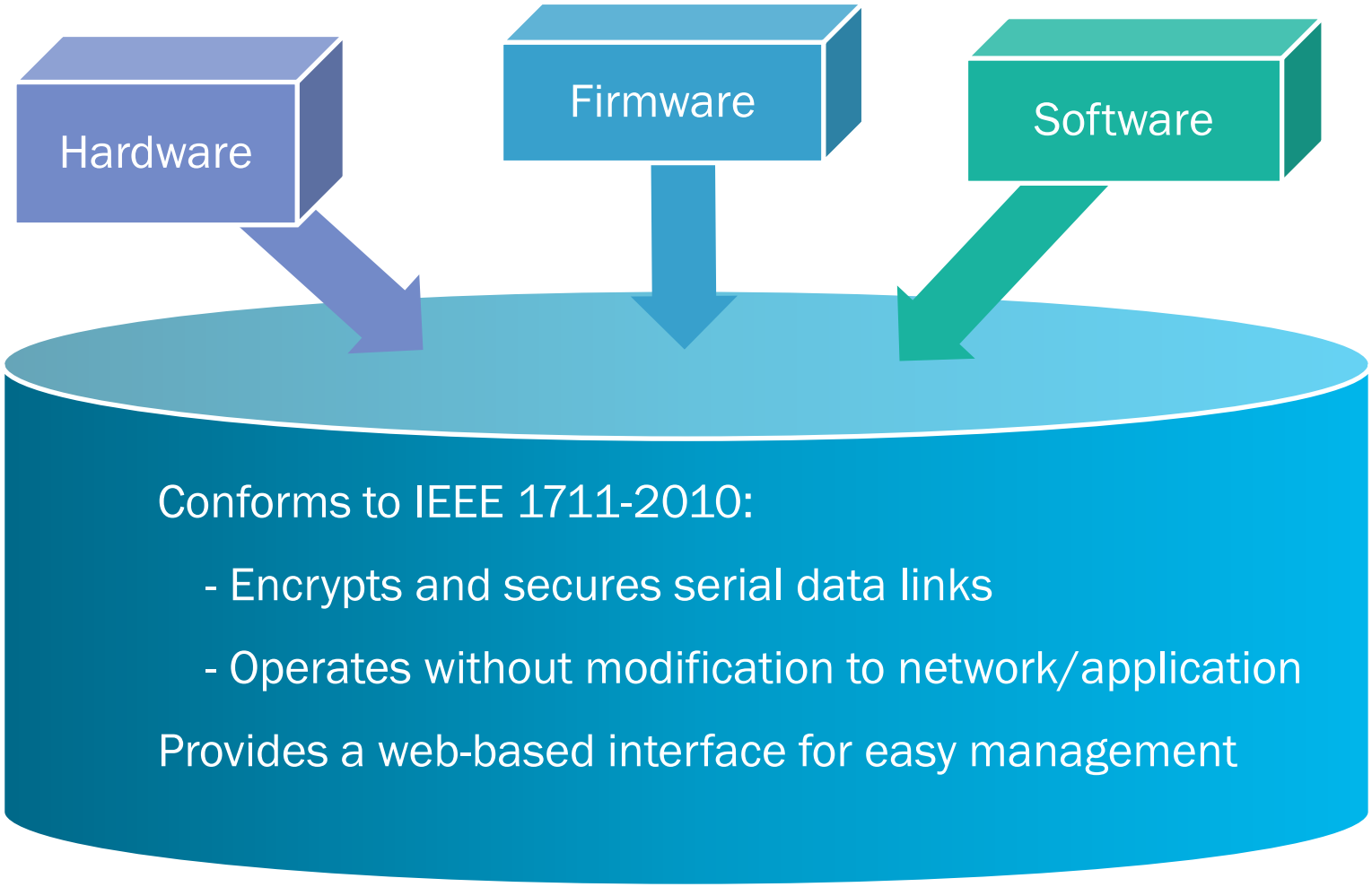
6.11 Legacy Related Risks

6.11.1 Untrusted and legacy devices and protocols - current biggest threat (KF 11.1)

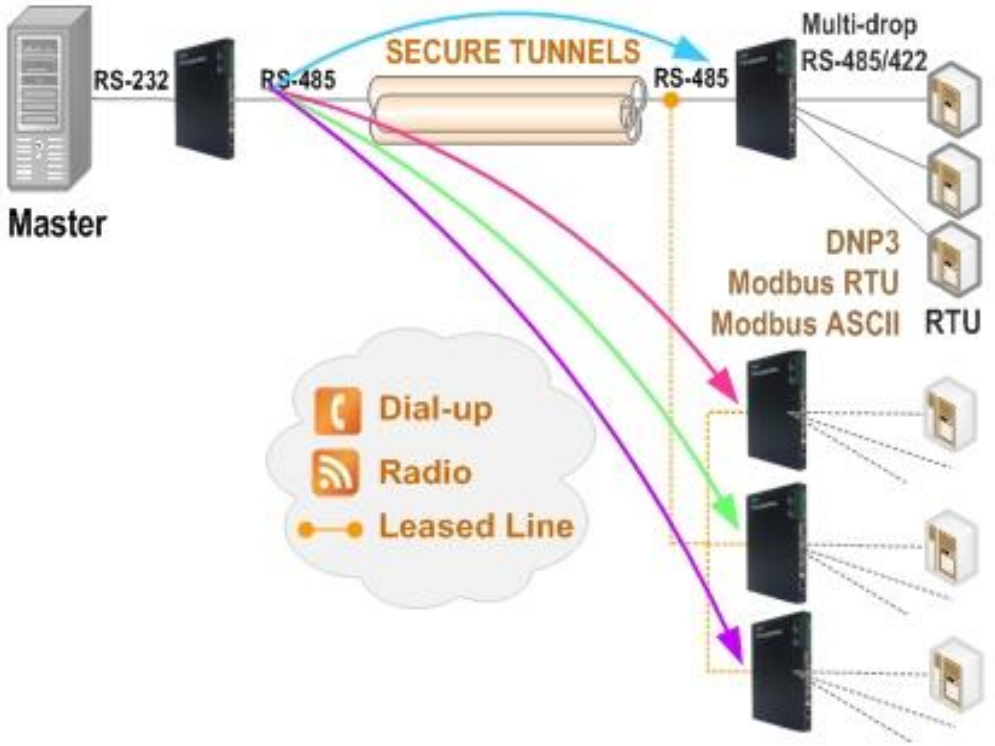
According to the survey, the biggest threat to the security of ICS is the existence of untrusted devices. This is usually related to the use of legacy or proprietary technologies that often include security breaches (e.g. backdoors).

Source: ENISA. *Protecting Industrial Control Systems: Recommendations for Europe and Member States.* December 19, 2011. http://www.enisa.europa.eu/act/res/other-areas/ics-scada/protecting-industrial-control-systems.-recommendations-for-europe-and-member-states-1

DESIGN OVERVIEW



DESIGN OVERVIEW – SYSTEM CONFIGURATION



- Up to 65,533 secure tunnels (IPSec-like)
- Each tunnel has its own Encryption Key & Authentication Key and protects one/more RTU

HARDWARE DESIGN - CRITERIA

µProcessor	• Not too fast and not too slow
Encryption	• Hardware co-processor*
I/O	• Ethernet • Three UARTs (RS-232/422/485)
Availability	• Product longevity (10+ yrs)
Development Tools / Support	• Mature and proven

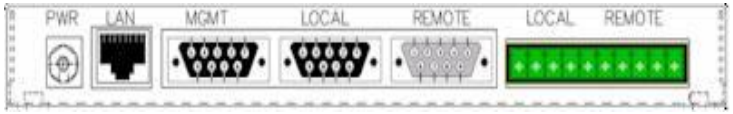
***Encryption delay :**

- Caused by block protocol encryption overhead, i.e., Header, Trailer, Message Authentication Code + Encryption processing
- The delay impact is greater for small messages

HARDWARE DESIGN – EMI GUIDELINES

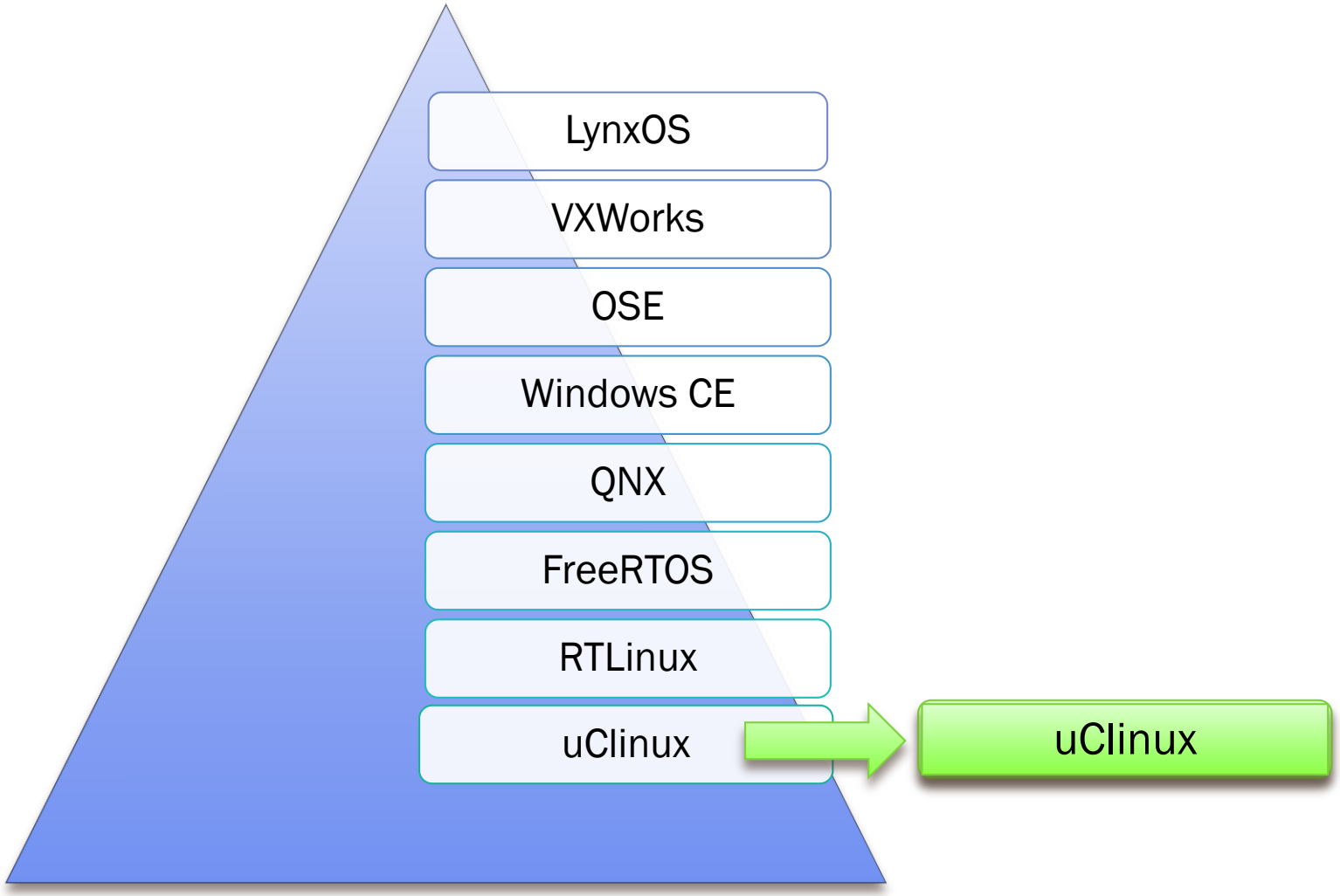
- **Controlled area**
  - Parts with very fast rise times, and those that are thermally hot, are noisy, or are high voltage
  - Signal timing considerations, such as differential pairs, critical clock signals, etc
  - Signals that might need to be guard banded
- **Power/Ground gridding**
  - High voltage and/or high current traces
- **Impedance controlled nets and their terminations**
  - 75||100 Ohm
- **Bypass caps**

HARDWARE DESIGN - CONNECTORS



Port Name	Connectors
Ethernet (LAN)	RJ-45
Management	RS-232
Local	RS-232/422/485
Remote	RS-232/422/485
Local/Remote	RS-422/485 Terminal Block 10-pin

FIRMWARE DESIGN - OPERATING SYSTEM



FIRMWARE DESIGN - UCLINUX

Pros

- Full Linux 2.6 kernel
- Built-in IP connectivity, file systems, applications…
- Lightweight (under 300KB)
- Faster than Linux (context switches: no cache flushes)
- Royalty free

Limitation

- No memory protection (no MMU)

FIRMWARE DESIGN – CODING & DEBUGGING

- uClinux
  - Initialization/boot loader tailored to processor
  - Kernel config to fit requirements
- Add–ins:
  - RTAI (Real Time Application Interface) - allows applications with strict timing constraints
  - Encryption drivers
  - Speed buffering
  - Modem emulation AT command set
- IEEE 1711 functionality

FIRMWARE DESIGN - IEEE 1711 STATE MACHINE EXAMPLE

Table 3 Session state machine for one dynamic session

Action / Event	Current State			
	closed	wait_ACK	wait_BEG	open
send OPN	wait_ACK, 3	X	X	wait_ACK, 3
send DTA	X	X	X	open, 1
send CLS	X	X	X	closed, 1
rcv OPN	wait_BEG, 5	wait_BEG, 4	wait_BEG, 8	wait_BEG, 11
rcv ACK	closed, 1	open, 6	wait_BEG, 1	open, 1
rcv BEG	closed, 1	wait_ACK, 1	open, 9	open, 1
rcv DTA	closed, 7	wait_ACK, 1	wait_BEG, 1	open, 2
rcv CLS	closed, 1	wait_ACK, 1	wait_BEG, 1	closed, 1
rcv ERR	closed, 1	closed, 10	closed, 9	closed, 1
rcv bad	closed, 1	wait_ACK, 1	wait_BEG, 1	open, 1
ACK timeout	X	closed, 1	X	X
BEG timeout	X	X	closed, 1	X

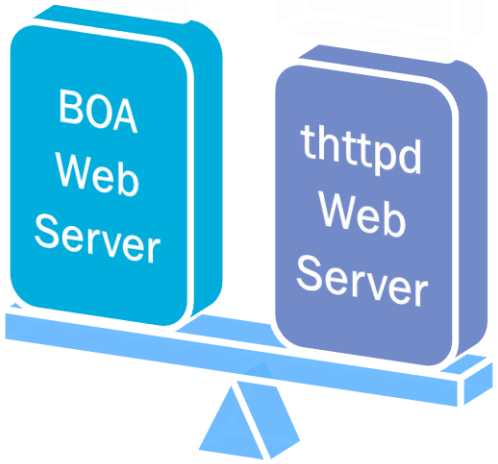
Note: X = cannot occur

An IEEE 1711 compliant implementation shall perform the following actions specified in Table 3:

1: do nothing
2: process payload
3: start ACK timer
4: cancel ACK timer, send ACK, start BEG timer
5: send ACK, start BEG timer
6: cancel ACK timer, send BEG
7: send ERR
8: cancel BEG timer, send ACK, start BEG timer
9: cancel BEG timer
10: cancel ACK timer
11: close current session D, send ACK, start BEG timer

```
/*******************************************************
    Session State Machine
    S+ = f(event, state)    and action = f(event, state)

              __State_              ___State_
     event |  S+          event | action

*******************************************************/
void session_state (void)
{
    // Execute action = f(event, state)
    switch (ucActionTable[ucEvent][ucState]) {
    case DO_NOTHING:              // action 1 of Table 3
        break;

    case PROCESS_PAYLOAD:         // action 2
        process_payload();
        break;

    case START_ACK_TIMER:         // action 3
        start_ACK_timer();
        break;

    case SEND_ACK:                // action 4
        cancel_ACK_timer();
        send_ACK();
        start_BEG_timer();
        break;

    // action 5 ... action 11

    default:
        break;
    }

    //  Update next state: state = f(event, state)
    ucState = ucStateTable[ucEvent][ucState];
}
```

SOFTWARE DESIGN - WEB SERVER

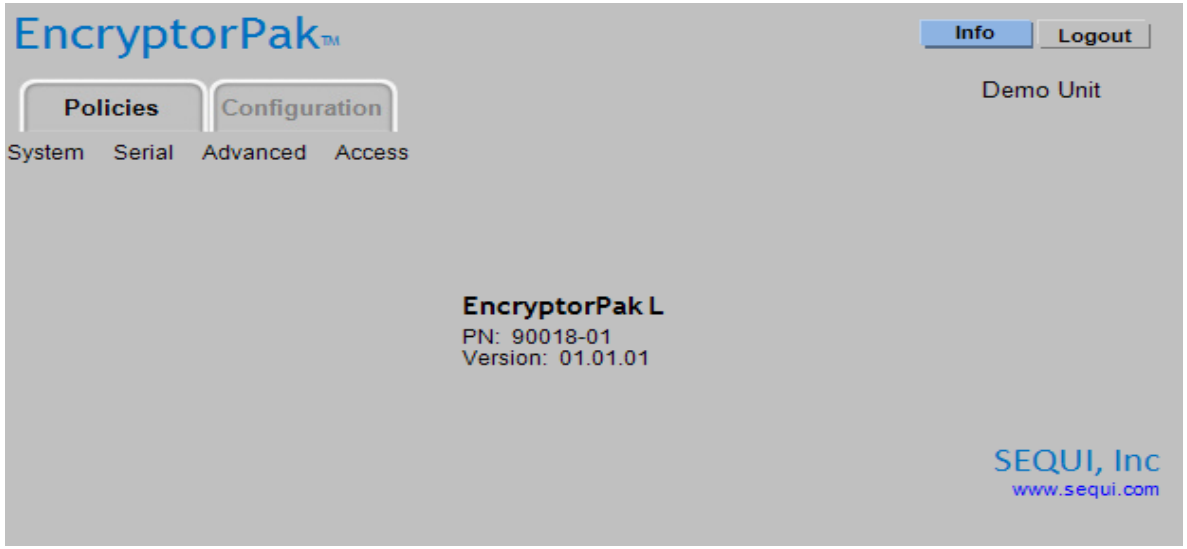


BOA and thttpd:

- Small, lightweight, and fast
- Popular among embedded crowd, esp. for embedded Linux

SOFTWARE DESIGN – USER INTERFACE

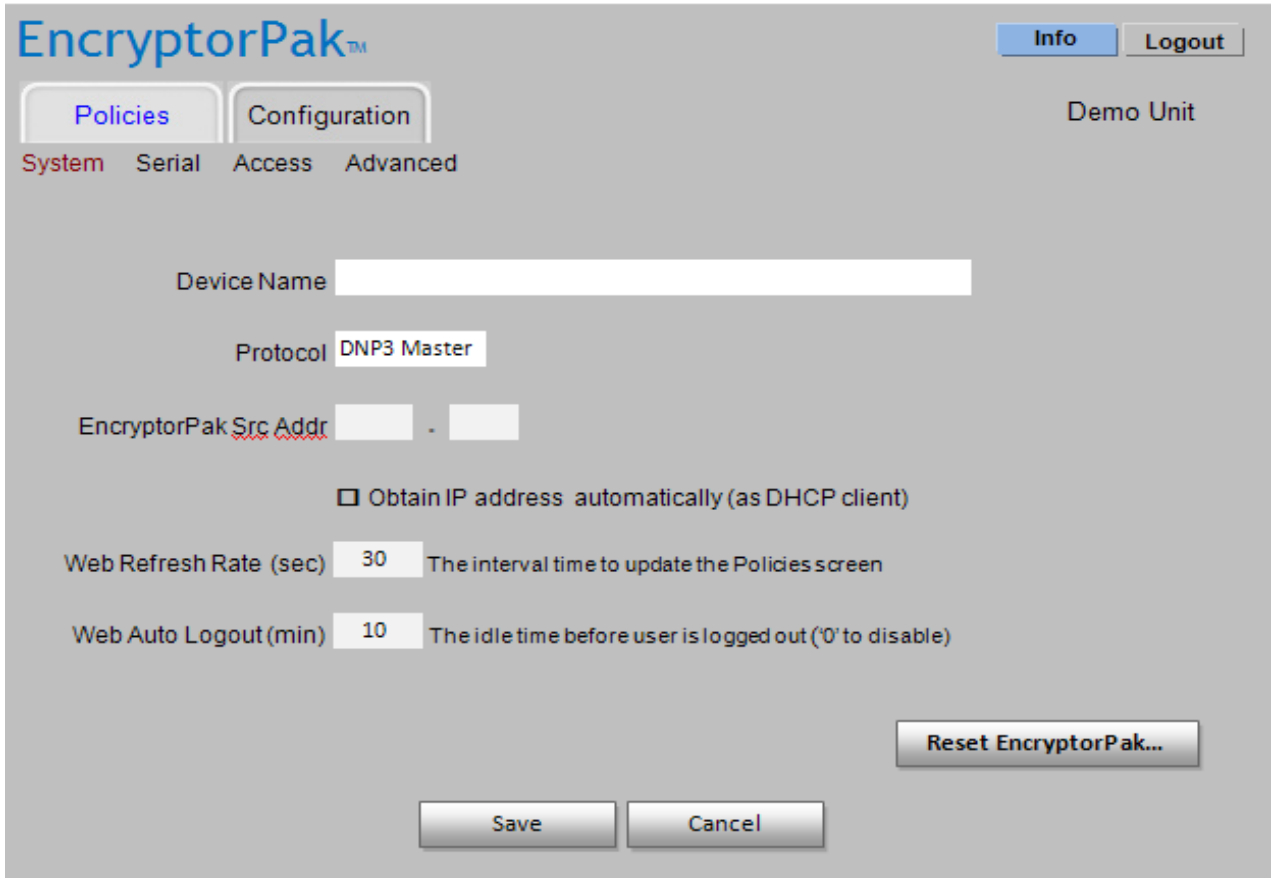
Configuration



- **System** – Configure operating parameters
- **Serial** – Configure port and data communications settings
- **Access** – Add user login accounts
- **Advanced** – Configure SCADA settings, cipher settings, and encryption

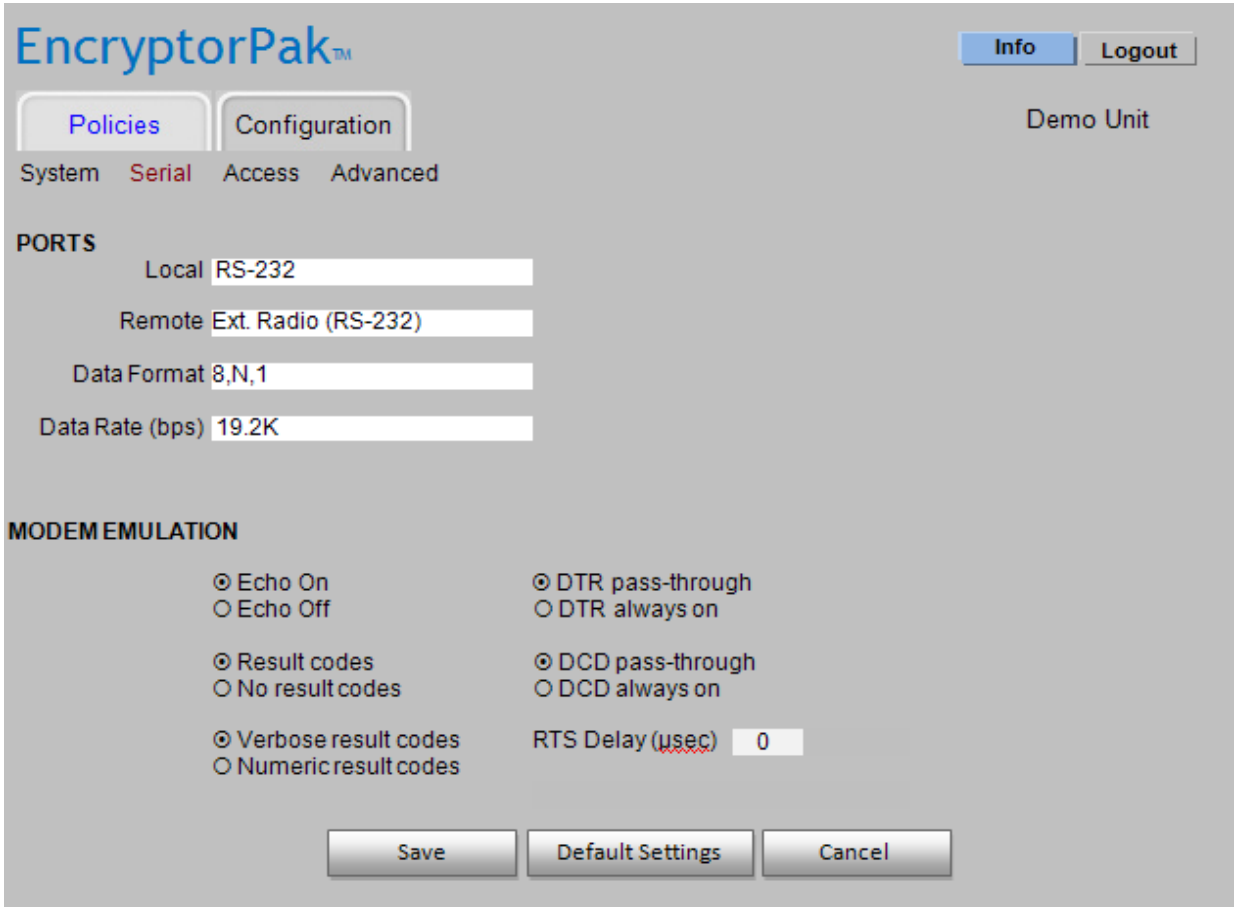
SOFTWARE DESIGN – USER INTERFACE

Configuration > System



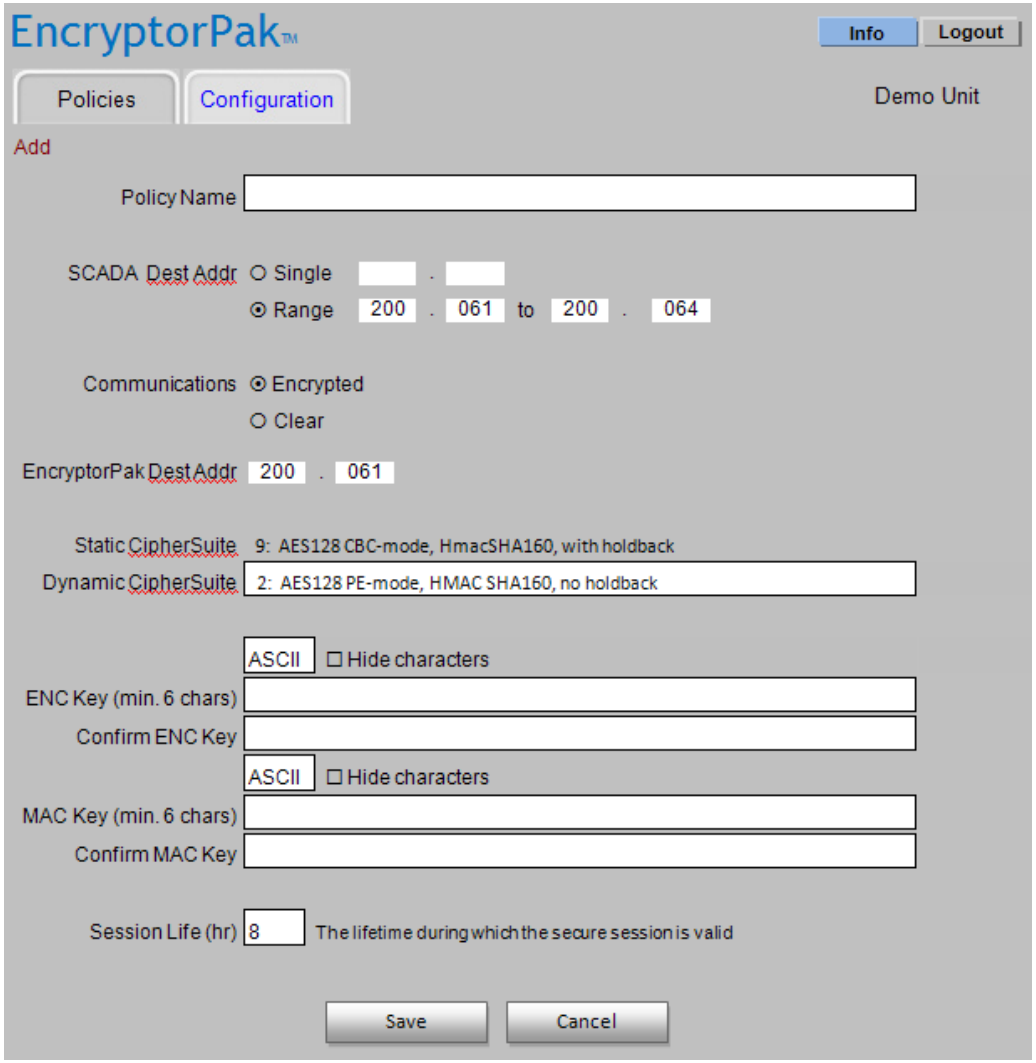
SOFTWARE DESIGN – USER INTERFACE

Configuration > Serial



SOFTWARE DESIGN – USER INTERFACE

Policies > Add



LEGACY SYSTEMS "GOING ETHERNET"

- The trend is to integrate serial SCADA protocols with corporate network for effective management and real-time business decisions

  However:

  - Serial protocols remain insecure, lack authentication (they're simply wrapped inside TCP/IP packets!)
  - The backdoor risk is still there
  - TCP/IP has its own vulnerabilities (that are widely shared within the computer underground!)

  Numerous FREE scanning, vulnerability discovery, and attack tools are available, such as *nmap, TCPview, Nessus, Attacker Tool Kit (ATK), Sniffit, Netcat, Wireshark*

  Visit "Top 100 Network Security Tools" at http://sectools.org/

IEEE 1711-2010 ENHANCEMENTS

Examples of vendor-added functionality:

- Management functions , such as *audit logs*, *reports*…
- Secure *Serial-over-Ethernet*
- Dial-up access control via *session negotiation*
- Interface to *low-cost wireless*, such as ZigBee® IEEE 802.15.4
- Custom embedded analog/digital remote I/O and data acquisition

ENHANCEMENTS – AUDIT LOGS, REPORTS

The table below describes pages of the Web interface and their access level permissions.

NAVIGATION PAGE	DESCRIPTION	ACCESS LEVEL		
		Administrator [1]	Crypto Officer	User [2]
Configuration	Clicking the tab opens the Product Information page	✓	✓	✓
System	Settings for protocol, source address, IP address, Web timers, and reset button	✓	✗	✗
Serial	Settings for serial ports	✓	✓	✓
Access	Setup login accounts	✓	✓ [3]	✓ [3]
Advanced	Settings for timeout, prebuffer, broadcast mode, and advanced cipher settings	✓	✓	✗
Policies	Clicking the tab opens the Policies Table page	✓	✓	✓
Add	Add a policy	✓	✓	✗
Copy	Copy a policy	✓	✓	✗
Edit	Edit a policy	✓	✓	✗
Delete	Delete policies	✓	✓	✗
DiagUp	Establish connection to remote device	✓	✓	✓
DiagDn	Terminate connection to remote device	✓	✓	✓
☐ [4]	Selected policy is in effect when point-to-point link is established	✓	✓	✗
Audit	Opens the Audit page for a summary of EncryptorPak activity	✓	✗	✗
Info	Opens the Info page for a quick view of EncryptorPak configuration settings	✓	✓	✓
Logout	Returns to the Login page	✓	✓	✓

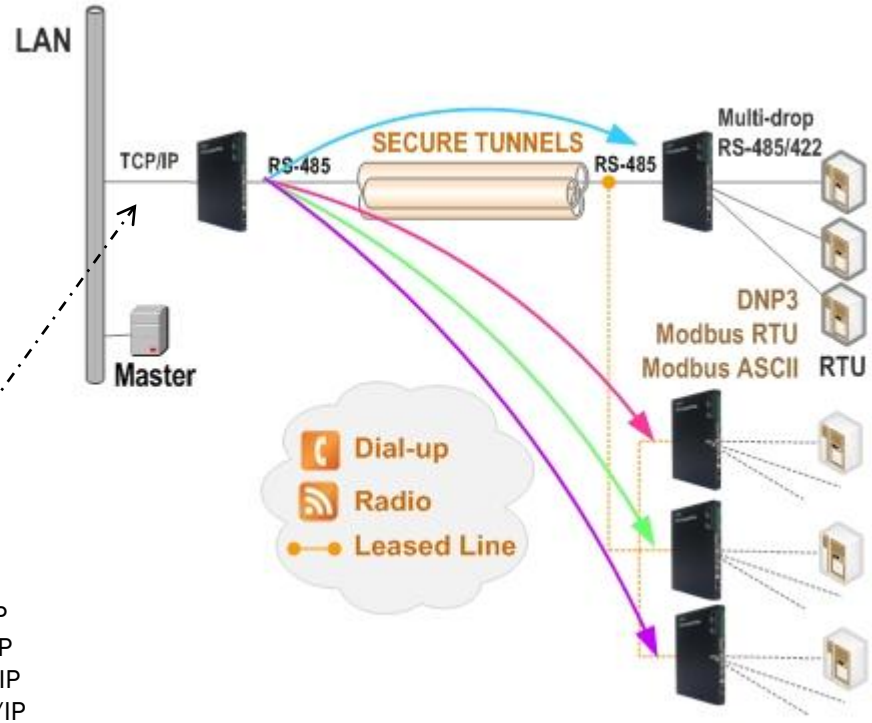
[1] *The Administrator level can be assigned to more than one login user*
[2] *The User level is intended primarily for users with testing responsibilities*
[3] *Access page only allows changing own password*
[4] *Available on menu bar only when Point-to-Point protocol is selected*

✓ Allowed
✗ Denied

Table 4. Description of Navigation Pages and Access Level Permissions

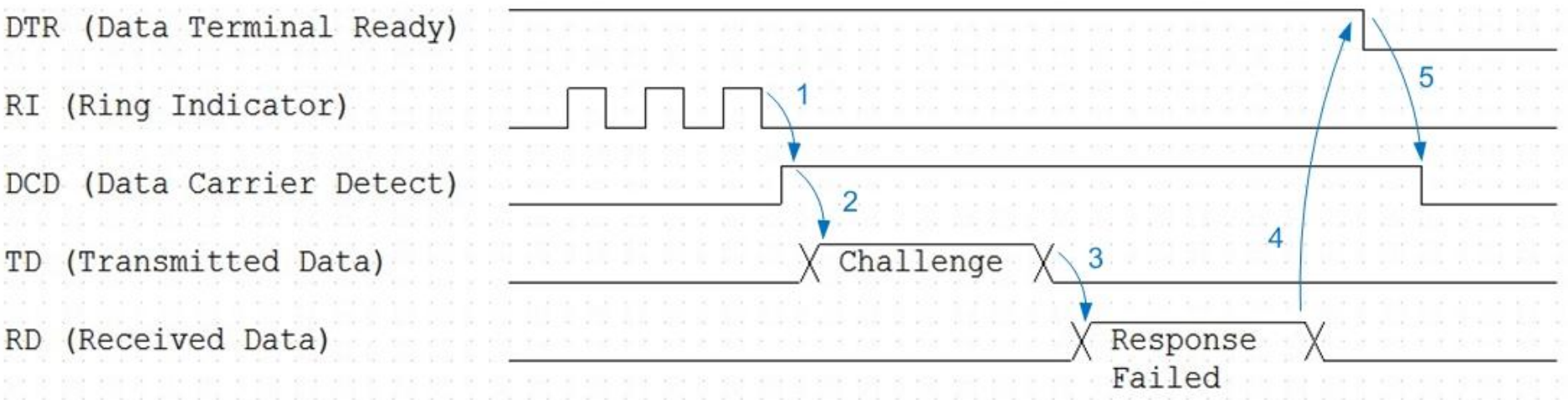
ENHANCEMENTS – SECURE SERIAL-OVER-ETHERNET

Modbus TCP/IP
Modbus UDP/IP
Modbus RTU Over TCP/IP
Modbus RTU Over UDP/IP
Modbus ASCII Over TCP/IP
Modbus ASCII Over UDP/IP
DNP3 over TCP/IP



ENHANCEMENTS – DIAL-UP ACCESS CONTROL

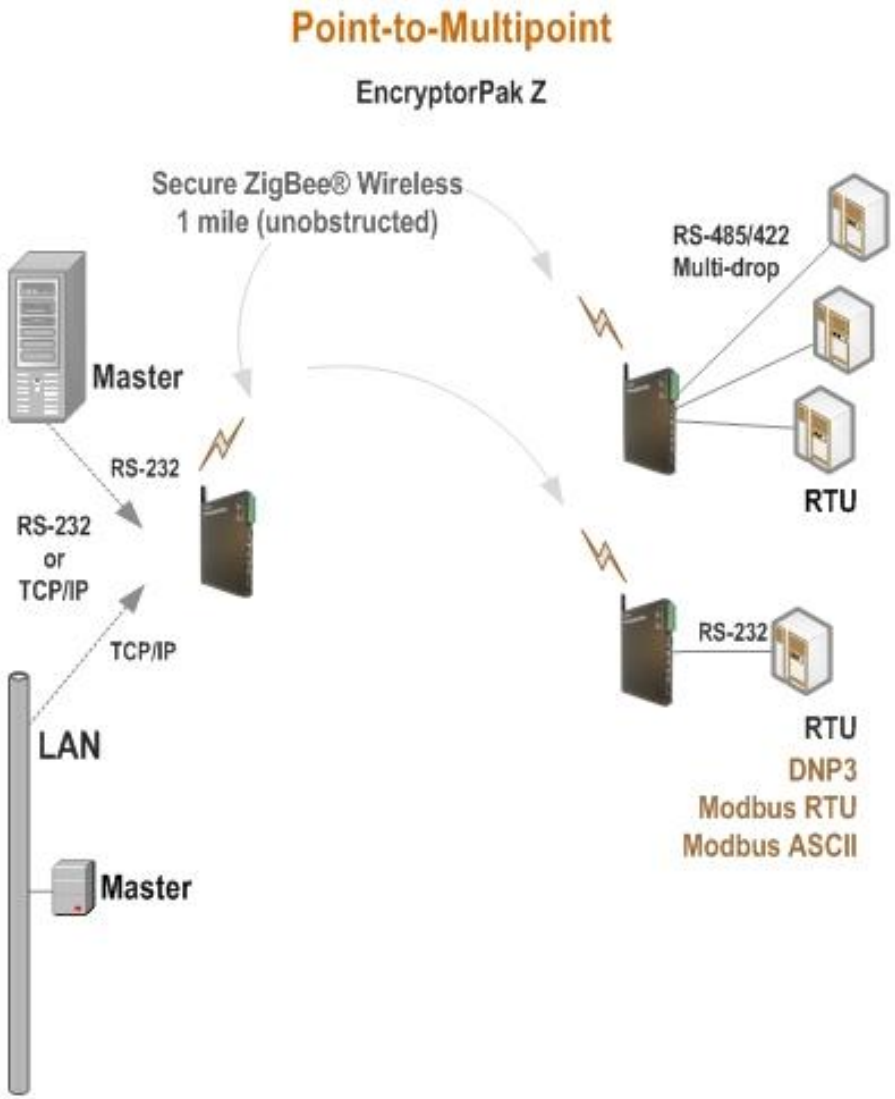
Timing diagram for answering device:



1. Receives an incoming call (RI toggle) and auto-answers (DCD active)
2. Transmits challenge data
3. Receives response data
4. If “failed” challenge, drop DTR
5. Modem disconnects call (DCD inactive)

ENHANCEMENTS – LOW-COST WIRELESS

ZigBee® Wireless



THANK YOU

Please feel free to send your comments or questions

Tien Van tvan@sequi.com
Tracy Amaio, Ph.D. teamaio@sequi.com