

# **Protecting Vulnerable Applications with IIS7**



# IIS7 Overview

- Modular Design
- Two Modes of Operation
  - Integrated Mode

New mode where the ASP.NET Runtime is integrated with the core web server
  - Compatibility Mode

Made for legacy support with previous version of IIS



# Leveraging Integrated Mode

- Unified Request Processing Pipeline
  - Exposed to both native and managed components (known as modules)
  - Native and managed modules to apply to all requests, regardless of handler
  - ASP.NET HttpModules comparable to ISAPI
  - Hook into one or more IIS7 pipeline events



# Protecting Applications on IIS7

- Modules participate in the processing of every request
  - Monitor, Change or Add to it
  - .NET class that implements the ASP.NET `System.Web.IHttpModule` interface
- Can perform processing tasks to secure inbound and outbound data
  - Request/Response Parsing
  - Data Validation
  - Data Encryption



# IIS7 Protection Module (SPF)

- HttpModule to Protect IIS7 Web Applications
  - **Input Tampering**  
Query String, Cookies, Form Inputs
  - **URI Tampering**  
Forceful Browsing
  - **Forgery, Hijacking / Cross-Site Attacks**  
Session-Based Tokenization



# IIS7 Protection Module

- Parses outbound HTML response
  - Links
  - Form Action URL
  - Form Inputs
  - Headers (Location, Set-Cookie)
- All embedded data is encrypted/tokenized and validated on subsequent requests
  - Machine Key Encryption (Cookies & Form Inputs)
  - SHA1 HMAC Token (URLs & Query Strings)



# SPF Cryptography

- Encrypted Data Elements
  - 16 Random Bytes Pre-Pended to Each Value
    - Produces Unique Cipher Text (similar to IV)
  - Uses ASP.NET Machine Key to Encrypt
  - Appends Time Stamp & SHA1 HMAC
    - Cipher Text
    - Time Stamp
    - Source IP Address
    - Session ID (Cookie)



# SPF Cryptography

- Tokenized Elements
  - Appends Time Stamp & SHA1 HMAC
    - Plain-Text (URL, Query String)
    - Time Stamp
    - Source IP Address
    - Session ID (Cookie)



# Configuration Options

- Default Protection Settings (Required)
  - Protection Scope (URI, QueryString, Forms, Cookies)
  - Main Application Entry Point (URL)
  
- Exceptions to Default Settings (Optional)
  - Global Exceptions (Form Inputs, Cookies)
  - URI Exceptions (URI, Query String, Form Inputs)



# Configuration Options

- Real-Time Tracking of Response Form Elements
  - Ineligible form inputs are automatically added to exception list (i.e. HTML text box)
  - Auto-generated exceptions can be dumped from memory
- “Black List” Regular Expression Filter
  - Used to optionally block malicious input strings



# Configuration Options

- Two Modes of Operation
  - Passive: Silent logging of all failed requests
  - Active: Failed requests are rejected
- Javascript Argument Protection (manual)
  - Javascript arguments used to pass request parameters can be encrypted like form inputs
    - `__doPostBack(eventTarget, eventArgument)`



# Live Demonstration

# DEMO



# Questions

- GDS Website:
  - <http://www.gdssecurity.com>
- GDS Blog:
  - <http://www.gdssecurity.com/l/b>