SECURITY EXTENSIONS TO JTAG

Linda Lieu
SecureHardware LLC
November 12, 2010
OUTLINE

- Background
- Problem Definition
- Possible Attacks
- Proposed Solutions
- Ongoing Work
**History of JTAG**

- Testing of multi-layer boards became much more difficult

- Bed-of-nails testing

- Joint Test Action Group (JTAG)

- IEEE Standard 1149.1 Test Access Port and Boundary Scan Architecture
JTAG STANDARD

- Increased controllability, faster testing time, lower cost

**Fig. 1** Printed circuit board with four chips

**Fig. 2** Boundary Scan Testing on a Chip
**JTAG Applications**

- Debugging of software and programming of integrated circuits
- Widespread use in telecommunications, medical, government, financial industry etc.
- Enabled in most devices
  - Mobile phones, computers, microwaves
WHAT IS AT RISK?

- Mission data
- Intellectual property
- Proper system functionality
JTAG Attacks

Source: Cisco Systems
POTENTIAL ATTACKERS

- Vendors of on-board components
- Producers of system components
- Users and/or consumers of the system
GOALS OF THE Attacker

- Disable or corrupt the system
- Extract embedded code or crypto key
- Embed trojan functionality without detection
- Duplicate system design
THREAT MODEL
ATTACKER OBTAINS TEST DATA

STANDARD COMPUTER WITH JTAG INTERFACE

DEVICE 1

DEVICE 2 (ATTACKER)

DEVICE 3 (VICTIM)

SECRET DATA FROM TESTER TO DEVICE 3

ATTACKER KEEPS A COPY OF DATA

VICTIM GETS UNMODIFIED DATA
TESTER RECEIVES FALSE RESPONSES

- BOGUS TEST RESULTS RETURNED TO TESTER

- STANDARD COMPUTER WITH JTAG INTERFACE (VICTIM)

- TEST VECTORS FROM TESTER TO DEVICE 3

- ATTACKER KEEPS A COPY OF DATA

- VICTIM GETS PUT INTO BYPASS MODE
ATTACKER READS OUT SECRET

ATTACKER 2 COLLECTS TEST RESPONSES FROM VICTIM

STANDARD COMPUTER WITH JTAG INTERFACE

DEVICE 5 (ATTACKER 2)

DEVICE 4

DEVICE 3 (VICTIM)

ATTACKER 1 INITIATES JTAG COMMUNICATION

VICTIM Responds Normally to Test Vectors
ATTACK DEMO
ATTACK DEMO
**OUR SOLUTION**

We propose the following security mechanisms:

- Device/chip identification
- Test data encryption
- Communication integrity
CHIP AUTHENTICATION

<table>
<thead>
<tr>
<th>CHAL</th>
<th>RESP1</th>
<th>RESP2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x23af4...</td>
<td>0xa938c...</td>
<td>0x32893...</td>
</tr>
<tr>
<td>0xeb4d8...</td>
<td>0x2459e...</td>
<td>0xf83bb...</td>
</tr>
<tr>
<td>0xa9c3e...</td>
<td>0xc9a5b...</td>
<td>0xb982c...</td>
</tr>
<tr>
<td>0x6d917...</td>
<td>0x9df44...</td>
<td>0xd1d8d...</td>
</tr>
<tr>
<td>0x0d0a3...</td>
<td>0x098a5...</td>
<td>0xfa083...</td>
</tr>
<tr>
<td>0x3b5c1...</td>
<td>0xda671...</td>
<td>0xc99b6...</td>
</tr>
<tr>
<td>0xd10c9...</td>
<td>0xe6333...</td>
<td>0x84d1a...</td>
</tr>
</tbody>
</table>
ENCRYPTION

- Secure JTAG Implementation
- Test PC
- session state

<table>
<thead>
<tr>
<th>CHAL</th>
<th>RESP1</th>
<th>RESP2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x23af4...</td>
<td>0xa938c...</td>
<td>0x32893...</td>
</tr>
<tr>
<td>0xeb4d8...</td>
<td>0x2459e...</td>
<td>0xf83bb...</td>
</tr>
<tr>
<td>0xa9c3e...</td>
<td>0xc9f8b...</td>
<td>0xb982c...</td>
</tr>
<tr>
<td>0x6d917...</td>
<td>0x9df44...</td>
<td>0x1d1d8d...</td>
</tr>
<tr>
<td>0x0d0a3...</td>
<td>0x098a5...</td>
<td>0xfa083...</td>
</tr>
<tr>
<td>0x3b5c1...</td>
<td>0xda671...</td>
<td>0xc99b6...</td>
</tr>
<tr>
<td>0xd10c9...</td>
<td>0xe6333...</td>
<td>0x84d1a...</td>
</tr>
</tbody>
</table>
ENCRYPTION

Secure JTAG Implementation

Test PC

session state

plaintext

plaintext
MESSAGE AUTHENTICATION CODE

- **Validate MAC**
  - Secure JTAG Implementation
  - ciphertext
  - ciphertext

- **Compute MAC**
  - If MAC is valid

- **Test PC**
  - MAC (DATA)
  - 32-bit IDCODE
CONTINUING WORK

- Identify other JTAG vulnerabilities
- Reduce impact on performance
- Explore different MACs and encryption algorithms
- Commercialize Secure JTAG design