A Taxonomy for Attack Patterns on Information Flows in Component-Based Operating Systems

Michael Hanspach\textsuperscript{1} Jörg Keller\textsuperscript{2}

\textsuperscript{1}Fraunhofer FKIE, Wachtberg, Germany
\textsuperscript{2}University Hagen, Germany

7th Layered Assurance Workshop, December 2013
1. Introduction
2. Basic Scenario
3. Information Flows
4. Attack Patterns
5. Conclusions
component-based operating systems threatened by illegitimate information flows (e.g. the MILS architectural approach)

- attack patterns based on shared resources, colluding components or physical characteristics
- confidentiality and integrity targeted by analysis
Contribution

1. simple algebra describing information flows in operating systems
2. taxonomy of attack patterns on these information flows (all possible information flows targeted)
3. identification of a new covert channel category: covert physical channel
Operating system model

Figure: Scenario for attacks on a component-based operating system

- \( n \geq 2 \) isolated application partitions
- virtual machines or isolated applications
- pure isolation hypervisors are targeted
Information Flows

Graph of information flows

- $P = \{p_0, \ldots, p_{n-1}\}$ represents isolated application partitions
- $K$ represent operating system kernel
- $H$ represent underlying hardware.
- $A$ is the attacker

![Graph of information flows](image-url)
Legitimate Information Flows

\[ \iff \text{is defined as bidirectional direct information flow} \]

1. \( p_i \iff K \)
2. \( K \iff H \)
   
   may be concatenated to:

3. \( p_i \iff K \iff H \)
Recursive Walk Algorithm

```plaintext
function walk-to-neighbors (start, walk)

for all neighbors of start do

    local-walk := concatenate(walk, neighbor)
    walk-legitimate := test-walk(start, neighbor)

    if (walk-legitimate = true)
        then
            walk-to-neighbors(neighbor, local-walk)
        else
            store local-walk
        end-if

end-for

end-function
```
Recursive Walk Algorithm

- start at $p_i$
- algorithm stops if information flow identified as illegitimate
- illegitimate information flows can be extended
- append $\leftrightarrow p_j$ to demonstrate channel between $p_i$ and $p_j$
- append $\leftrightarrow A$ to demonstrate an attack
- algorithm verified with Perl program
Illegitimate Information Flows (Basic Scenario)

1. $p_i \iff A$
2. $p_i \iff p_j$
3. $p_i \iff H$
4. $p_i \iff K \iff A$
5. $p_i \iff K \iff p_j$
6. $p_i \iff K \iff H \iff p_j$
7. $p_i \iff K \iff H \iff A$
Scenario with operating system guards

- using a trustworthy middleware (e.g. the MILS architectural approach)
Legitimate Information Flows (Extended Scenario)

1. $c_k \iff K \iff H$
2. $g_k \iff K \iff c_k$
3. $p_i \iff K \iff g_k$
4. $g_k \iff K \iff H$

concatenated information flows:
5. $p_i \iff K \iff g_k \iff K \iff c_k \iff K \iff H$
6. $p_i \iff K \iff g_k \iff K \iff H$

illegitimate access to a component:
7. $p_i \iff K \iff c_k \iff K \iff p_j$
Attack Patterns

- information flows mapped to attack patterns
- map information flows to attack patterns
- attack patterns: specific methods for establishment of illegitimate information flows
- information flows: attack patterns as a many-to-many-relation
- multiple patterns mapped to a single information flow
- multiple information flows mapped to a single attack pattern
## Attack patterns mapped to information flows

<table>
<thead>
<tr>
<th>No.</th>
<th>Information Flow</th>
<th>Basic Attack Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$p_i \iff A$</td>
<td>Use insider attack</td>
</tr>
<tr>
<td>2</td>
<td>$p_i \iff p_j$</td>
<td>Exploit flaws in the system policy</td>
</tr>
<tr>
<td>3</td>
<td>$p_i \iff H$</td>
<td>Use physical attacks, Exploit covert channels, Exploit storage channels</td>
</tr>
<tr>
<td>4</td>
<td>$p_i \iff K \iff A$</td>
<td>Exploit flaws in the system policy</td>
</tr>
<tr>
<td>5</td>
<td>$p_i \iff K \iff p_j$</td>
<td>Exploit covert channels, Exploit flaws in the system policy</td>
</tr>
<tr>
<td>6</td>
<td>$p_i \iff K \iff H \iff p_j$</td>
<td>Use physical attacks, Exploit covert channels</td>
</tr>
<tr>
<td>7</td>
<td>$p_i \iff K \iff H \iff A$</td>
<td>Use physical attacks</td>
</tr>
<tr>
<td>8</td>
<td>$p_i \iff K \iff c_k \iff K \iff p_j$</td>
<td>Exploit flaws in the system policy</td>
</tr>
</tbody>
</table>

**Table:** Illegitimate information flows mapped to basic attack patterns
## Basic and extended attack patterns

<table>
<thead>
<tr>
<th>No.</th>
<th>Basic Attack Pattern</th>
<th>Extended Attack Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Use insider attack</td>
<td>N/A</td>
</tr>
<tr>
<td>2</td>
<td>Use physical attacks</td>
<td>Exploit physical access, Exploit side channels</td>
</tr>
<tr>
<td>3</td>
<td>Exploit implementation flaws</td>
<td>N/A</td>
</tr>
<tr>
<td>4</td>
<td>Exploit covert channels</td>
<td>Exploit covert storage channels, Exploit covert timing channels, Exploit covert physical channels</td>
</tr>
<tr>
<td>5</td>
<td>Exploit storage channels</td>
<td>N/A</td>
</tr>
<tr>
<td>6</td>
<td>Exploit flaws in the system policy</td>
<td>Exploit corrupt policy channels, Exploit illegitimate access to components</td>
</tr>
</tbody>
</table>

**Table:** Basic attack patterns mapped to extended attack patterns
### Attack patterns and references

<table>
<thead>
<tr>
<th>No.</th>
<th>Attack Pattern</th>
<th>Description</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Use insider attack</td>
<td>A is granted direct access to extract information from $p_i$.</td>
<td>Baracaldo and Joshi, Liu et al., Yu and Chiueh.</td>
</tr>
<tr>
<td>2</td>
<td>Exploit physical access</td>
<td>A has physical access to the computing system and can directly extract information, which has been stored at $H$, from $p_i$.</td>
<td>Halderman et al.</td>
</tr>
<tr>
<td>3</td>
<td>Exploit side channel</td>
<td>A uses side channel attacks to extract information from $p_i$ over $H$.</td>
<td>van Eck, Dümuth, Backes et al., LeMay and Tan, Shamir and Tromer, Halevi and Saxena.</td>
</tr>
<tr>
<td>4</td>
<td>Exploit implementation flaws</td>
<td>A uses implementation flaws of $p_i$, $K$ or $H$ to extract information from $p_i$.</td>
<td>Klein et al., Boettcher et al., Robinson et al.</td>
</tr>
<tr>
<td>5</td>
<td>Exploit covert storage channel</td>
<td>A covert channel is established between $p_i$ and $p_j$ by storing a hidden message within shared resources.</td>
<td>Lampson, National Computer Security Center.</td>
</tr>
<tr>
<td>6</td>
<td>Exploit covert timing channel</td>
<td>A covert channel is established between $p_i$ and $p_j$ by encoding a message via the timing behavior of shared resources.</td>
<td>Lampson, National Computer Security Center.</td>
</tr>
<tr>
<td>7</td>
<td>Exploit covert physical channel</td>
<td>A covert physical channel is established between $p_i$ and $p_j$ by encoding a message, sending it out into the physical environment via $H$ and receiving it at a different interface of $H$.</td>
<td>N/A.</td>
</tr>
<tr>
<td>8</td>
<td>Exploit storage channels</td>
<td>Information can be exchanged between $p_i$ and $p_j$ via shared storage resources.</td>
<td>Lampson</td>
</tr>
<tr>
<td>9</td>
<td>Exploit corrupt policy channel</td>
<td>Information can be exchanged directly between $p_i$ and $p_j$ due to a flawed MAC policy.</td>
<td>Agreiter, Zhai et al.</td>
</tr>
<tr>
<td>10</td>
<td>Exploit illegitimate access to components</td>
<td>Information can be exchanged between $p_i$ and $p_j$ by exploiting illegitimate access to a component.</td>
<td>N/A.</td>
</tr>
</tbody>
</table>

**Table:** Description and references regarding to attack patterns
A new type of covert channel

- concept of a *covert physical channel*
- uses physical signals
- transmits covert messages between isolated application partitions
- not a CPC: covert storage channel based on disk arm optimization (Karger and Wray)
- introducing a variable $E$ (describing the shared physical environment)
- disk arm channel described by: $p_i \iff K \iff H \iff K \iff p_j$
- covert physical channel described by:
  
  $p_i \iff K \iff H \iff E \iff H \iff K \iff p_j$
- not a CPC: covert channel based on clock skew manipulations as result of different heat output levels (Murdoch)
- signal is not directly transferred between sender and receiver over shared physical environment
Attack patterns for physical communication

Figure: Physical Attack Patterns
Examples for physical communication

1. a covert physical channel
2. covert acoustical mesh network (multi-hop, 20 bit/s up to 19.7 m)  
   (Hanspach and Goetz, Journal of Communications)
Offtopic: Covert Acoustical Mesh Network Experiment

Figure: Schematic view of the interconnection experiment at Fraunhofer FKIE
Offtopic: A closer look at the audio filtering guard

Figure: An audio filtering guard used by two different application partitions
Suggested Taxonomy

Category

Basic Attack Patterns
- Use insider attacks
- Use physical attacks
- Exploit implementation flaws

Extended Attack Patterns
- Exploit physical access
- Exploit side channel attack
- Exploit covert storage channel
- Exploit covert timing channel
- Exploit covert physical channel
- Exploit corrupt policy channel
- Exploit illegitimate access to components

Read or modify data without proper authorization
- Exploit covert channels
- Exploit storage channels
- Exploit implementation flaws
- Use physical attacks
- Use insider attacks

Figure: Taxonomy of attack patterns in component-based operating systems
Conclusions

- systematic study of information flows in operating system models
- every possible information flow covered
- information flows are mapped to specific attack patterns
- more detailed attack patterns might be developed in future
- algebra might be used for future information flow analyses
- covert physical channels are introduced (more work coming soon...) 
- attack patterns might be used to systematically harden operating systems