Heap-based buffer overflow on *nix

You might need to actively scan your Linux hosts for indicators of a successfully run heap-based buffer overflow (Baron Samedit attack) when doing the following:

- Detecting the Sudo Baron Samedit attack

Prerequisites

In order to execute this procedure in your environment, the following data, services, or apps are required:

- Splunk Enterprise or Splunk Cloud Platform
- *nix logs
- Add-on for OSquery

Example

Your organization has unpatched Linux servers. You know those servers are vulnerable to a number of attacks, including the heap-based buffer overflow in sudo, called Baron Samedit. You want to be able to search for indications that this attack has hit your servers.

To optimize the searches shown below, you should specify an index and a time range.

Option 1

1. Run the following search:
   "sudoedit -s "
   | stats values(sourcetype) values(source) values(host)

Search explanation

The table provides an explanation of what each part of this search achieves. You can adjust this query based on the specifics of your environment.

<table>
<thead>
<tr>
<th>Splunk Search</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;sudoedit -s &quot;</td>
<td>Look for this string in your *nix logs.</td>
</tr>
<tr>
<td><strong>Splunk Search</strong></td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td></td>
</tr>
<tr>
<td>stats values(sourcetype) values(source) values(host)</td>
<td></td>
</tr>
<tr>
<td><strong>Explanation</strong></td>
<td></td>
</tr>
<tr>
<td>List values that show the source of the string.</td>
<td></td>
</tr>
</tbody>
</table>

**Option 2**

1. Install the Add-on for OSquery.
2. Add the following line to the OSquery TA props.conf:
   ```
   FIELDALIAS-process = columns.cmdline as process
   ```
3. Run the following search:
   ```
   sourcetype="osquery:results"
   eventtype="osquery-process"
   | search process="sudoedit -s \*"
   ```

**Search explanation**

The table provides an explanation of what each part of this search achieves. You can adjust this query based on the specifics of your environment.

<table>
<thead>
<tr>
<th><strong>Splunk Search</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>sourcetype=&quot;osquery:results&quot;</td>
</tr>
<tr>
<td>eventtype=&quot;osquery-process&quot;</td>
</tr>
<tr>
<td>search process=&quot;sudoedit -s *&quot;</td>
</tr>
<tr>
<td><strong>Explanation</strong></td>
</tr>
<tr>
<td>Search only osquery:results logs.</td>
</tr>
<tr>
<td>Search only for osquery-process events.</td>
</tr>
<tr>
<td>Look for events where the process is set to sudoedit -s , which is an indicator that the vulnerability has been exploited.</td>
</tr>
</tbody>
</table>

**Result**

If the above search produces results, the exploit has been run on the machine(s) that are listed in the result. If any are found, the next step is to respond to the incident. A suitable response is to isolate the machine and look for signs of lateral movement, data exfiltration, or credential dumping. The attacker would have elevated privileges to root or the super user and could do almost anything from that point on.

If no results are found, either the exploit has not been used or coverage is incomplete. The Splunk Samedit TA will search your *nix machines for un-patched versions and can output an inventory of machines that have the vulnerability. That information can be used to further investigate malicious activity and to provide a list of machines that need to be patched.