User Right Test Proposal

# What are the use cases for the proposed test? Why is it needed?

The proposed userright\_test is an alternative way to collect user right/privilege information on Windows operating systems. The existing accesstoken\_test collects user right/privilege information, but it does so in a way that is not scalable in certain situations (e.g., domain controllers).

For example, to collect the users who have the privilege SE\_BACKUP\_NAME using the existing accesstoken\_test, all security principles (users, groups, etc.) must be enumerated on the target system. For each security principle found, all user rights/privileges (not just SE\_BACKUP\_NAME) must be collected, even if the security principle does not have the user right/privilege. This is usually done by using the pattern match operation on the security\_principle object entity with a value of “.\*”. This can be an expensive operation, especially on domain controllers with many users. Also, the generated OVAL results can contain many accesstoken\_items that can be unwieldy.

The proposed userright\_test would hopefully alleviate some of this burden by collecting only the security principles that actually have the specified user right/privilege.

# What OVAL Component schema(s) are affected?

The following OVAL component schema(s) are affected:

* windows-definitions-schema.xsd
* windows-system-characteristics-schema.xsd

# What will the OVAL Object construct look like?

The following userright\_object will collect all security principles that have been given the SE\_BACKUP\_NAME privilege. A scanner could use the [LsaEnumerateAccountsWithUserRight](http://msdn.microsoft.com/en-us/library/windows/desktop/ms721792(v=vs.85).aspx) function to collect the security principles that have been granted the user right/privilege.

<userright\_object id="oval:test:obj:1" version="1">

<userright>SE\_BACKUP\_NAME</userright>

</userright\_object>

The userright object entity holds a string that represents the name of a particular user right/privilege. The string value is restricted to the enumerated types defined in [appendix A](#_Appendix_A_–). If the xsi:nil attribute is used, then all accounts in the LSA database of the target system that have any user right/privilege assigned to them will be enumerated (to accomplish this, a scanner could pass NULL in the UserRights parameter for the [LsaEnumerateAccountsWithUserRight](http://msdn.microsoft.com/en-us/library/windows/desktop/ms721792(v=vs.85).aspx) function). There are no restrictions on operations for the userright entity.

# What will the OVAL State construct look like?

The following userright\_state will check that only Backup Operators or Administrators have a given privilege.

<userright\_state id="oval:test:ste:1" version="1">

<security\_principle operation="pattern match" entity\_check="all">^(([Bb][Aa][Cc][Kk][Uu][Pp] [Oo][Pp][Ee][Rr][Aa][Tt][Oo][Rr][Ss])|([Aa][Dd][Mm][Ii][Nn][Ii][Ss][Tt][Rr][Aa][Tt][Oo][Rr][Ss]))$</security\_principle>

</userright\_ state>

The security\_principle state entity holds a string that identifies the user or group that has been granted the specified user right/privilege. Security principles include users or groups with either local or domain accounts, and computer accounts created when a computer joins a domain. In Windows, security principles are case-insensitive. As a result, it is recommended that the case-insensitive operations are used for this entity. User rights and permissions to access objects such as Active Directory objects, files, and registry settings are assigned to security principles. In a domain environment, security principles should be identified in the form: "domain\trustee name". For local security principles use: "computer name\trustee name". For built-in accounts on the system, use the trustee name without a domain. The security\_principle element can be included multiple times in a system characteristic item in order to record that a user right/privilege has been granted to a number of users/groups. Note that the entity\_check attribute associated with EntityStateStringType guides the evaluation of entities like security\_principle that refer to items that can occur an unbounded number of times.

# What will the OVAL Item construct look like?

The following userright\_item shows how security principles with the SE\_BACKUP\_NAME privilege would be collected. Note that multiple security\_principle entities can be collected in a single item.

<userright\_item status="exists" id="1">

<userright>SE\_BACKUP\_NAME</userright>

<security\_principle>Backup Operators</security\_principle>

<security\_principle>Administrators</security\_principle>

</userright\_ item >

As an alternative to collecting all security\_principle within a single userright\_item, a userright\_item could be created for each security\_principle entity. This may be required if set/filter logic is needed. The above examples rely on the entity\_check attribute of the security\_principle state entity to come to a result.

The userright item entity holds a string that represents the name of a particular user right/privilege. The string value is restricted to the enumerated types defined in [appendix A](#_Appendix_A_–). There are no restrictions on operations for the userright entity.

The security\_principle item entity holds a string that identifies the user or group that has been granted the specified user right/privilege. Multiple security\_principle item entities can exist within a single userright\_item. Security principles include users or groups with either local or domain accounts, and computer accounts created when a computer joins a domain. In Windows, security principles are case-insensitive. As a result, it is recommended that the case-insensitive operations are used for this entity. User rights and permissions to access objects such as Active Directory objects, files, and registry settings are assigned to security principles. In a domain environment, security principles should be identified in the form: "domain\trustee name". For local security principles use: "computer name\trustee name". For built-in accounts on the system, use the trustee name without a domain. The security\_principle element can be included multiple times in a system characteristic item in order to record that a user right/privilege has been granted to a number of users/groups. Note that the entity\_check attribute associated with EntityStateStringType guides the evaluation of entities like security\_principle that refer to items that can occur an unbounded number of times.

# Appendix A – Userright Entity Item Values

The following table contains the values that can be used in the userright entity item for OVAL objects, states and items.

| **“userright” Entity Item Value** | **Description** |
| --- | --- |
| SE\_ASSIGNPRIMARYTOKEN\_NAME | This privilege is required to assign the primary token of a process. |
| SE\_AUDIT\_NAME | This privilege is required to generate audit-log entries. |
| SE\_BACKUP\_NAME | This privilege is required to perform backup operations. |
| SE\_CHANGE\_NOTIFY\_NAME | This privilege is required to receive notifications of changes to files or directories. |
| SE\_CREATE\_GLOBAL\_NAME | This privilege is required to create named file mapping objects in the global namespace during Terminal Services sessions. |
| SE\_CREATE\_PAGEFILE\_NAME | This privilege is required to create a paging file. |
| SE\_CREATE\_PERMANENT\_NAME | This privilege is required to create a permanent object. |
| SE\_CREATE\_SYMBOLIC\_LINK\_NAME | This privilege is required to create a symbolic link. |
| SE\_CREATE\_TOKEN\_NAME | This privilege is required to create a primary token. |
| SE\_DEBUG\_NAME | This privilege is required to debug and adjust the memory of a process owned by another account. |
| SE\_ENABLE\_DELEGATION\_NAME | This privilege is required to mark user and computer accounts as trusted for delegation. |
| SE\_IMPERSONATE\_NAME | This privilege is required to impersonate. |
| SE\_INC\_BASE\_PRIORITY\_NAME | This privilege is required to increase the base priority of a process. |
| SE\_INCREASE\_QUOTA\_NAME | This privilege is required to increase the quota assigned to a process. |
| SE\_INC\_WORKING\_SET\_NAME | This privilege is required to allocate more memory for applications that run in the context of users. |
| SE\_LOAD\_DRIVER\_NAME | This privilege is required to load or unload a device driver. |
| SE\_LOCK\_MEMORY\_NAME | This privilege is required to lock physical pages in memory. |
| SE\_MACHINE\_ACCOUNT\_NAME | This privilege is required to create a computer account. |
| SE\_MANAGE\_VOLUME\_NAME | This privilege is required to enable volume management privileges. |
| SE\_PROF\_SINGLE\_PROCESS\_NAME | This privilege is required to gather profiling information for a single process. |
| SE\_RELABEL\_NAME | This privilege is required to modify the mandatory integrity level of an object. |
| SE\_REMOTE\_SHUTDOWN\_NAME | This privilege is required to shut down a system using a network request. |
| SE\_RESTORE\_NAME | This privilege is required to perform restore operations. |
| SE\_SECURITY\_NAME | This privilege is required to perform a number of security-related functions, such as controlling and viewing audit messages. |
| SE\_SHUTDOWN\_NAME | This privilege is required to shut down a local system. |
| SE\_SYNC\_AGENT\_NAME | This privilege is required for a domain controller to use the Lightweight Directory Access Protocol directory synchronization services. |
| SE\_SYSTEM\_ENVIRONMENT\_NAME | This privilege is required to modify the nonvolatile RAM of systems that use this type of memory to store configuration information. |
| SE\_SYSTEM\_PROFILE\_NAME | This privilege is required to gather profiling information for the entire system. |
| SE\_SYSTEMTIME\_NAME | This privilege is required to modify the system time. |
| SE\_TAKE\_OWNERSHIP\_NAME | This privilege is required to take ownership of an object without being granted discretionary access. |
| SE\_TCB\_NAME | This privilege identifies its holder as part of the trusted computer base. |
| SE\_TIME\_ZONE\_NAME | This privilege is required to adjust the time zone associated with the computer's internal clock. |
| SE\_TRUSTED\_CREDMAN\_ACCESS\_NAME | This privilege is required to access Credential Manager as a trusted caller. |
| SE\_UNDOCK\_NAME | This privilege is required to undock a laptop. |
| SE\_UNSOLICITED\_INPUT\_NAME | This privilege is required to read unsolicited input from a terminal device. |
| SE\_BATCH\_LOGON\_NAME | This account right is required for an account to log on using the batch logon type. |
| SE\_DENY\_BATCH\_LOGON\_NAME | This account right explicitly denies an account the right to log on using the batch logon type. |
| SE\_DENY\_INTERACTIVE\_LOGON\_NAME | This account right explicitly denies an account the right to log on using the interactive logon type. |
| SE\_DENY\_NETWORK\_LOGON\_NAME | This account right explicitly denies an account the right to log on using the network logon type. |
| SE\_DENY\_REMOTE\_INTERACTIVE\_LOGON\_NAME | This account right explicitly denies an account the right to log on remotely using the interactive logon type. |
| SE\_DENY\_SERVICE\_LOGON\_NAME | This account right explicitly denies an account the right to log on using the service logon type. |
| SE\_INTERACTIVE\_LOGON\_NAME | This account right is required for an account to log on using the interactive logon type. |
| SE\_NETWORK\_LOGON\_NAME | This account right is required for an account to log on using the network logon type. |
| SE\_REMOTE\_INTERACTIVE\_LOGON\_NAME | This account right is required for an account to log on remotely using the interactive logon type. |
| SE\_SERVICE\_LOGON\_NAME | This account right is required for an account to log on using the service logon type. |

# Appendix B – References

| **Reference** | **Description** |
| --- | --- |
| LsaEnumerateAccountsWithUserRight Function | This function was used as the basis for the user right test (userright\_test). For more information, visit <http://msdn.microsoft.com/en-us/library/windows/desktop/ms721792(v=vs.85).aspx> |
| LsaEnumerateAccountRights Function | This function was possibly used at the basis for the existing access token test (accesstoken\_test). For more information, visit <http://msdn.microsoft.com/en-us/library/windows/desktop/ms721790(v=vs.85).aspx> |