SDDL Demystified Mind Map

by Splunk Threat Research Team

Predfined SDDL Strings For Device Objects

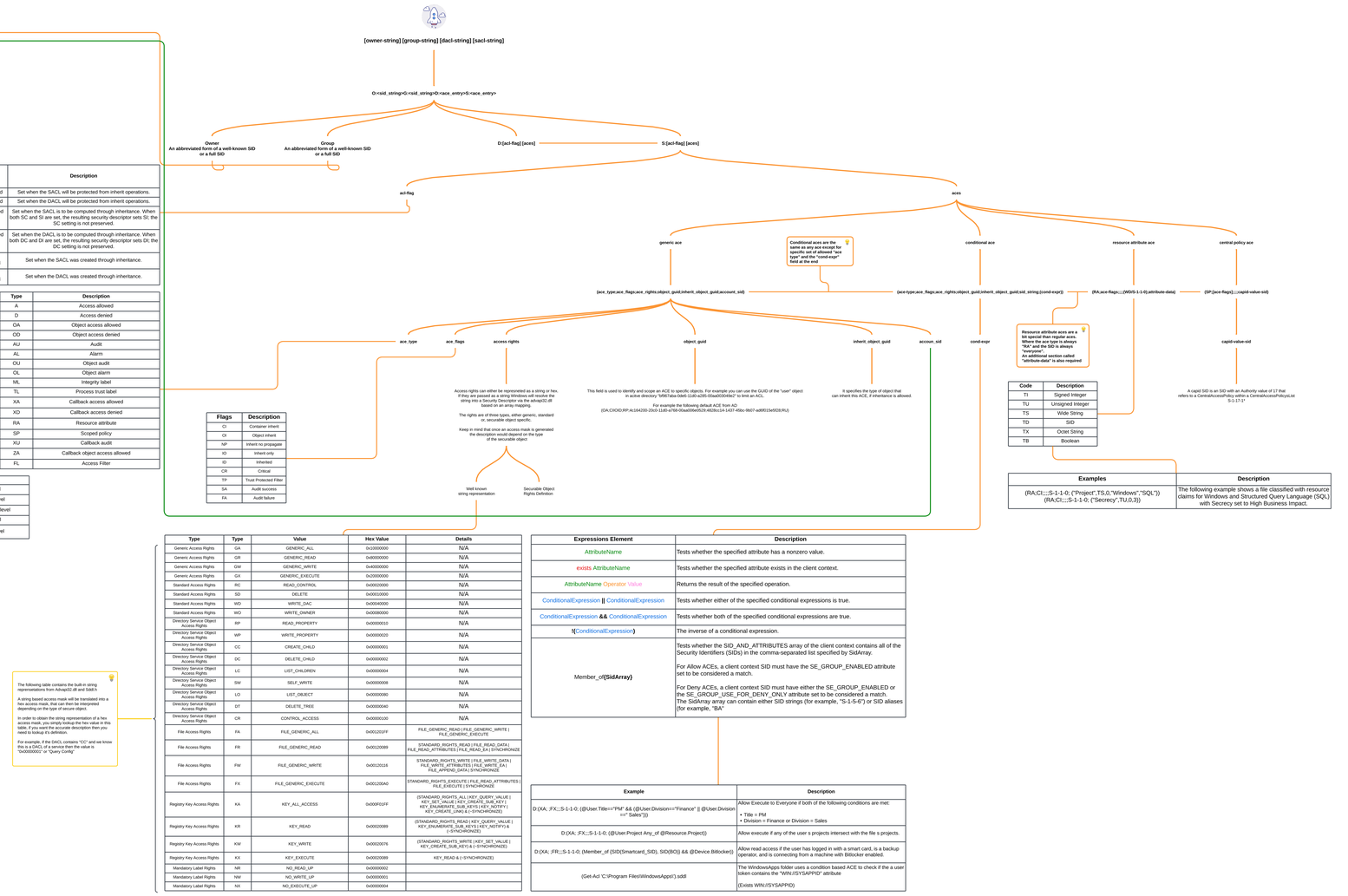
SDDL	SDDL String	Description
SDDL_DEVOBJ_KERNEL_ONLY	D:P	See wdmsec.h for more details
SDDL_DEVOBJ_SYS_ALL	D:P(A;;GA;;;SY)	See wdmsec.h for more details
SDDL_DEVOBJ_SYS_ALL_ADM_ALL	D:P(A;;GA;;;SY)(A;;GA;;;BA)	Allows the kernel, system, and administrator complete control over the device. No other users may access the device.
SDDL_DEVOBJ_SYS_ALL_ADM_RWX_WORLD_R	D:P(A;;GA;;;SY)(A;;GRGWGX;;;BA)(A;;GR;;;WD)	Allows the kernel and system complete control over the device. By default the administrator can access the entire device, but cannot change the ACL (the administrator must take control of the device first.)
SDDL_DEVOBJ_SYS_ALL_ADM_RWX_WORLD_R_RES_R	D:P(A;;GA;;;SY)(A;;GRGWGX;;;BA)(A;;GR;;;WD)(A;;GR;;;RC)	See wdmsec.h for more details

Value	Description
AO	Account operators
PA	Group Policy administrators
RU	Alias to allow previous Windows 2000
IU	Interactively logged-on user
AN	Anonymous logon
LA	Local administrator
AU	Authenticated users
LG	Local guest
BA	Built-in administrators
LS	Local service account
BG	Built-in guests
SY	Local system
BO	Backup operators
NU	Network logon user
BU	Built-in users
NO	Network configuration operators
CA	Certificate server administrators
NS	Network service account
CG	Creator group
PO	Printer operators
CO	Creator owner
PS	Personal self
DA	Domain administrators
PU	Power users
DC	Domain computers
RS	RAS servers group
DD	Domain controllers
RD	Terminal server users
DG	Domain guests
RE	Replicator
DU	Domain users
RC	Restricted code This SID is used to control access by untrusted code. ACL validation against tokens with RC consists of two checks, one against the token's normal list of SIDs (containing WD for instance), and one against a second list (typically containing RC and a subset of the original token SIDs). Access is granted only if a token passes both tests. As such, RC actually works in combination with other SIDs. Any ACL that specifies RC must also specify WD. When RC is paired with WD in an ACL, a superset of Everyone including untrusted code is described. Untrusted code might be code launched using the Run As option in Explorer. By default, World does not cover untrusted code.
EA	Enterprise administrators
SA	Schema administrators
ED	Enterprise domain controllers
SO	Server operators
WD	Everyone
SU	Service logon user
AA	Access control assistant operators Members of this group can remotely query authorization attributes and permissions for resources on this computer.
AC	All applications running in an app package context
CD	Users who can connect to certification authorities using DCOM
CN	Members of this group that are domain controllers may be cloned.
CY	Crypto Operators
ER	Event log readers
ES	Endpoint servers Servers in this group run virtual machines and host sessions where users RemoteApp programs and personal virtual desktops run.
HA	Hyper-V administrators Members of this group have complete and unrestricted access to all features of Hyper-V.
IS	Anonymous Internet Users
LU	Performance Log Users
MS	Management servers Servers in this group can perform routine administrative actions on servers running Remote Desktop Services.
MU	Performance Monitor Users
OW	Owner Rights SID
RA	RDS remote access servers Servers in this group enable users of RemoteApp programs and personal virtual desktops access to these resources.
RM	Remote management users Members of this group can access WMI resources over management protocols (such as WS-Management via the Windows Remote Management service). This applies only to WMI namespaces that grant access to the user.
RO	Enterprise Read-only domain controllers
UD	UserMode driver This SID grants access to user-mode drivers. Currently, this SID covers only drivers that are written for the User-Mode Driver Framework (UMDF). This SID is available starting with Windows 8. In earlier versions of Windows, which do not recognize the "UD" abbreviation, you must specify the fully qualified form of this SID (S-1-5-84-0-0-0-0-0) to grant access to UMDF drivers.
WR	Write Restricted code
AS	Authentication Authority Asserted
SS	Authentication Service Asserted
AP	Protected users
KA	Domain key credential administrators
EK	Enterprise key credential administrators

Acl Flag	Security Descriptor Equivalent	Details	Description
P	PS	SACL Protected	Set when the SACL will be protected from inherit operations.
P	PD	DACL Protected	Set when the DACL will be protected from inherit operations.
AR	SC	SACL Computed (Inheritance Required)	Set when the SACL is to be computed through inheritance. When both SC and SI are set, the resulting security descriptor sets SI; the SC setting is not preserved.
AR	DC	DACL Computed (Inheritance Required)	Set when the DACL is to be computed through inheritance. When both DC and DI are set, the resulting security descriptor sets DI; the DC setting is not preserved.
AI	SI	SACL Auto-Inherited	Set when the SACL was created through inheritance.
AI	DI	DACL Auto-Inherited	Set when the DACL was created through inheritance.

Ace Type	Scope	Type	Description
Ace	DACL	A	Access allowed
Ace	DACL	D	Access denied
Ace	DACL	OA	Object access allowed
Ace	DACL	OD	Object access denied
Ace	SACL	AU	Audit
Ace	SACL	AL	Alarm
Ace	SACL	OU	Object audit
Ace	SACL	OL	Object alarm
Ace	DACL	ML	Integrity label
Ace	DACL	TL	Process trust label
Conditional Ace	DACL	XA	Callback access allowed
Conditional Ace	DACL	XD	Callback access denied
Resource Attribute Ace	SACL	RA	Resource attribute
Ace	DACL	SP	Scoped policy
Conditional Ace	SACL	XU	Callback audit
Conditional Ace	DACL	ZA	Callback object access allowed
Ace	DACL	FL	Access Filter

Value	Description
LW	Low mandatory level
ME	Medium mandatory level
MP	Medium Plus mandatory level
HI	High mandatory level
SI	System mandatory level



Flags	Description
CI	Container inherit
OI	Object inherit
NP	Inherit no propagate
IO	Inherit only
ID	Inherited
CR	Critical
TP	Trust Protected Filter
SA	Audit success
FA	Audit failure

Examples	Description
(RA;CI;;;S-1-1-0; ("Project",TS,0,"Windows","SQL")) (RA;CI;;;;S-1-1-0; ("Secrecy",TU,0,3))	The following example shows a file classified with resource claims for Windows and Structured Query Language (SQL) with Secrecy set to High Business Impact.

The following table contains the built-in string representations from Advapi32.dll and Sddl.h

A string based access mask will be translated into a hex access mask, that can then be interpreted depending on the type of secure object.

In order to obtain the string representation of a hex access mask, you simply lookup the hex value in this table, if you want the accurate description then you need to lookup it's definition.

For example, if the DACL contains "CC" and we know this is a DACL of a service then the value is "0x00000020" or "Query Config"

Type	Type	Value	Hex Value	Details
Generic Access Rights	GA	GENERIC_ALL	0x10000000	N/A
Generic Access Rights	GR	GENERIC_READ	0x80000000	N/A
Generic Access Rights	GW	GENERIC_WRITE	0x40000000	N/A
Generic Access Rights	GX	GENERIC_EXECUTE	0x20000000	N/A
Standard Access Rights	RC	READ_CONTROL	0x00020000	N/A
Standard Access Rights	SD	DELETE	0x00010000	N/A
Standard Access Rights	WD	WRITE_DAC	0x00040000	N/A
Standard Access Rights	WO	WRITE_OWNER	0x00080000	N/A
Directory Service Object Access Rights	RP	READ_PROPERTY	0x00000010	N/A
Directory Service Object Access Rights	WP	WRITE_PROPERTY	0x00000020	N/A
Directory Service Object Access Rights	CC	CREATE_CHILD	0x00000001	N/A
Directory Service Object Access Rights	DC	DELETE_CHILD	0x00000002	N/A
Directory Service Object Access Rights	LC	LIST_CHILDREN	0x00000004	N/A
Directory Service Object Access Rights	SW	SELF_WRITE	0x00000008	N/A
Directory Service Object Access Rights	LO	LIST_OBJECT	0x00000080	N/A
Directory Service Object Access Rights	DT	DELETE_TREE	0x00000040	N/A
Directory Service Object Access Rights	CR	CONTROL_ACCESS	0x00000100	N/A
File Access Rights	FA	FILE_GENERIC_ALL	0x001201FF	FILE_GENERIC_READ | FILE_GENERIC_WRITE | FILE_GENERIC_EXECUTE
File Access Rights	FR	FILE_GENERIC_READ	0x00120089	STANDARD_RIGHTS_READ | FILE_READ_DATA | FILE_READ_ATTRIBUTES | FILE_READ_EA | SYNCHRONIZE
File Access Rights	FW	FILE_GENERIC_WRITE	0x00120116	STANDARD_RIGHTS_WRITE | FILE_WRITE_DATA | FILE_WRITE_ATTRIBUTES | FILE_WRITE_EA | FILE_APPEND_DATA | SYNCHRONIZE
File Access Rights	FX	FILE_GENERIC_EXECUTE	0x001200A0	STANDARD_RIGHTS_EXECUTE | FILE_READ_ATTRIBUTES | FILE_EXECUTE | SYNCHRONIZE
Registry Key Access Rights	KA	KEY_ALL_ACCESS	0x000F01FF	(STANDARD_RIGHTS_ALL | KEY_QUERY_VALUE | KEY_SET_VALUE | KEY_CREATE_SUB_KEY | KEY_ENUMERATE_SUB_KEYS | KEY_NOTIFY | KEY_CREATE_LINK) & (~SYNCHRONIZE)
Registry Key Access Rights	KR	KEY_READ	0x00020089	(STANDARD_RIGHTS_READ | KEY_QUERY_VALUE | KEY_ENUMERATE_SUB_KEYS | KEY_NOTIFY) & (~SYNCHRONIZE)
Registry Key Access Rights	KW	KEY_WRITE	0x00020076	(STANDARD_RIGHTS_WRITE | KEY_SET_VALUE | KEY_CREATE_SUB_KEY) & (~SYNCHRONIZE)
Registry Key Access Rights	KX	KEY_EXECUTE	0x00020089	KEY_READ & (~SYNCHRONIZE)
Mandatory Label Rights	NR	NO_READ_UP	0x00000002	
Mandatory Label Rights	NW	NO_WRITE_UP	0x00000001	
Mandatory Label Rights	NX	NO_EXECUTE_UP	0x00000004	

Expressions Element	Description
AttributeName	Tests whether the specified attribute has a nonzero value.
exists AttributeName	Tests whether the specified attribute exists in the client context.
AttributeName Operator Value	Returns the result of the specified operation.
ConditionalExpression || ConditionalExpression	Tests whether either of the specified conditional expressions is true.
ConditionalExpression && ConditionalExpression	Tests whether both of the specified conditional expressions are true.
!(ConditionalExpression)	The inverse of a conditional expression.
Member_of{SidArray}	Tests whether the SID_AND_ATTRIBUTES array of the client context contains all of the Security Identifiers (SIDs) in the comma-separated list specified by SidArray. For Allow ACEs, a client context SID must have the SE_GROUP_ENABLED attribute set to be considered a match. For Deny ACEs, a client context SID must have either the SE_GROUP_ENABLED or the SE_GROUP_USE_FOR_DENY_ONLY attribute set to be considered a match. The SidArray array can contain either SID strings (for example, "S-1-5-67) or SID aliases (for example, "BA"

Example	Description
D:(XA;;FX;;;S-1-1-0; ((@User.Title=="PM" && (@User.Division=="Finance" || @User.Division=="Sales")))	Allow Execute to Everyone if both of the following conditions are met: • Title = PM • Division = Finance or Division = Sales
D:(XA;;FX;;;S-1-1-0; (@User.Project Any_of @Resource.Project))	Allow execute if any of the user s projects intersect with the file s projects.
D:(XA;;FR;;;S-1-1-0; (Member_of {SID(Smartcard_SID), SID(BO)) && @Device.Bitlocker))	Allow read access if the user has logged in with a smart card, is a backup operator, and is connecting from a machine with Bitlocker enabled.
(Get-Acl 'C:\Program Files\WindowsApps\').sddl	The WindowsApps folder uses a condition based ACE to check if the a user token contains the "WIN://SYSAPPID" attribute (Exists WIN://SYSAPPID)