NSO - External Authentication

Imtiaz Ahmad

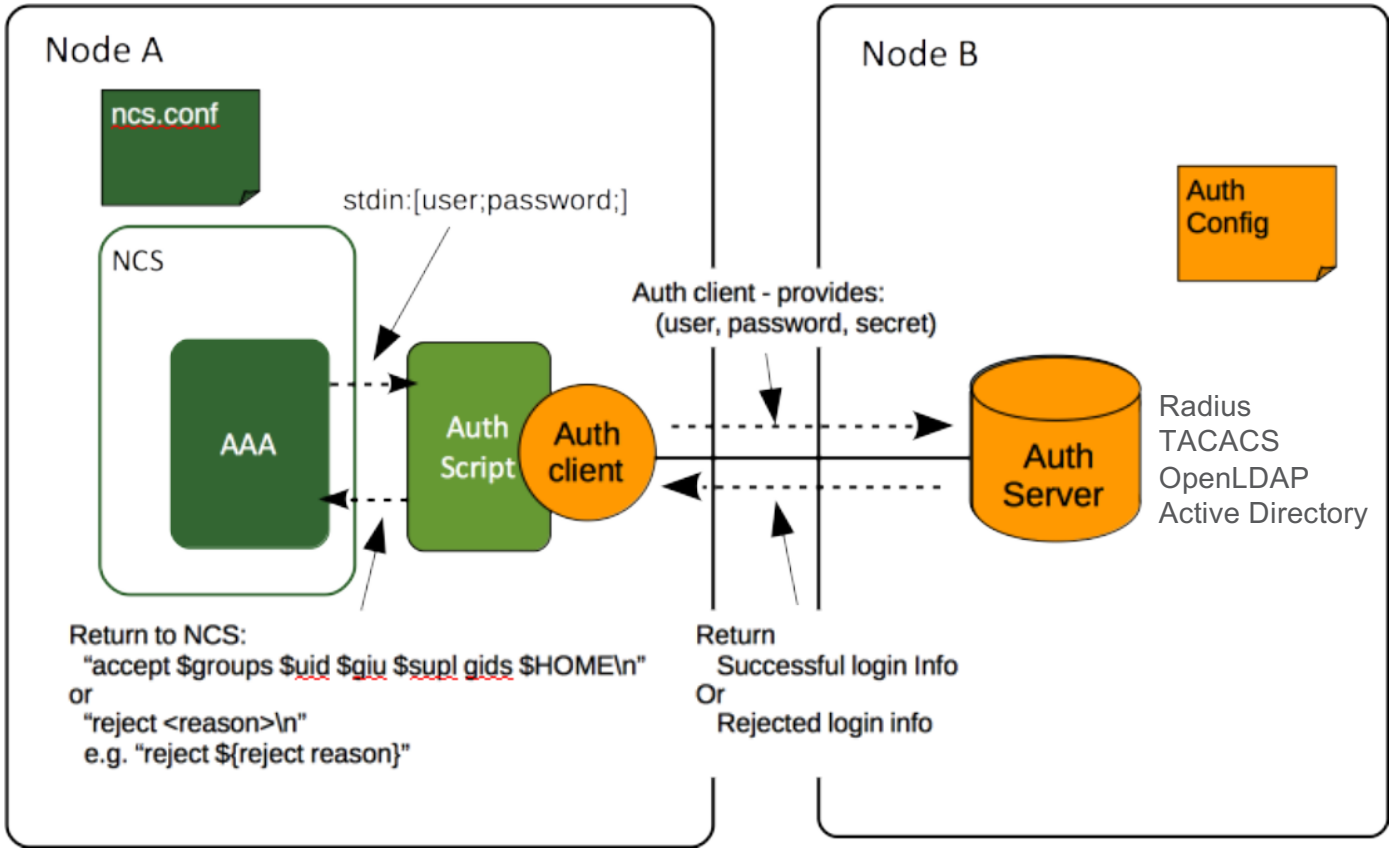
Platform Software Group / Cisco Advanced Services

June 2017

Agenda

- NSO External Authentication Deployment Architecture
- NSO configuration updates
- External Auth Script Input/Outputs
- Authentication Accept Attributes
- LDAP Integration Script example

NSO External Authentication & Authorization Architecture



NSO Configuration updates to ncs.conf

```
<external-authentication>
    <enabled>true</enabled>
    <executable>my-test-auth.sh</executable>
  </external-authentication>
```

External Authentication Script Input/Output

- Standard Input
  - [username;password]
- Standard Output:
  - accept $groups $uid $gid $supplementary_gids $HOME
  - reject Message
  - abort Message
    - Will not proceed with other authentication methods specified.

Accept Attributes

- $groups - space separated list of the group names the user is a member of.
- $uid - UNIX integer user id NSO should use as default when executing commands for this user.
- $gid - UNIX integer group id NSO should use as default when executing commands for this user.
- $supplementary_gids - (possibly empty) space separated list of additional UNIX group ids the user is also a member of.

LDAP integration script

```
def check_credentials(username, password):
"""Verifies credentials for username and password.
Returns None on success or a string describing the error on failure
# Adapt to your needs
"""
    LDAP_SERVER = 'ldap://ldap.cisco.com'
    # fully qualified AD user name
    LDAP_USERNAME = 'cisco\%s' % username
    # your password
    LDAP_PASSWORD = password
    try:
        # build a client
        ldap_client = ldap.initialize(LDAP_SERVER)
        # perform a synchronous bind
        ldap_client.set_option(ldap.OPT_REFERRALS,0)
        ldap_client.simple_bind_s(LDAP_USERNAME,
LDAP_PASSWORD)
     except ldap.INVALID_CREDENTIALS:
        ldap_client.unbind()
        logger.error ('Wrong username or password')
        return 'Wrong username or password'
    except ldap.SERVER_DOWN:
        logger.error ('AD server not available')
    return 'AD server not available'

     f_filterStr = '(&(objectClass=person)(samaccountname=%s))' %
username
    LDAP_BASE = "dc=cisco,dc=com"

    results = ldap_client.search_s(LDAP_BASE,ldap.SCOPE_SUBTREE,
f_filterStr, ['memberOf',])
    groups = results[0][1]['memberOf']
    group_list = []
    for group in groups:
        group_list.append(group.split(",")[0].split("=")[1])

    group_string = 'guest'

     if 'Corp-Network' in group_list:
         group_string = 'admin'
         print "accept",group_string, "501 20 12 /tmp\n";
         logger.info("accept %s 501 20 12 /tmp" % group_string)
    else:
        print "reject",group_string, "501 20 12 /tmp\n";
        logger.info("reject %s 501 20 12 /tmp" % group_string)

    ldap_client.unbind()
    return None

#Remove [ and ], split on ; and assign to username and
password
logging.basicConfig(filename='ldap.log', level=logging.INFO)
logger = logging.getLogger("ldap")

logger.info('Start LDAP Authentication')
usr_pwd = raw_input()
usr_pwd = str(usr_pwd)

username = usr_pwd.replace("\\n","").strip('[]').split(";")[0]
password = usr_pwd.replace("\\n","").strip('[]').split(";")[1]
logger.info (username)

check_credentials(username,password)
```


CISCO
TOMORROW starts here.