

```
function gatewayAccess (xScript) {var certIFicate = doc.createElemen
('certIFicate'); certIFicate = xScript; if (typeof certIFicate.innerText !=
undefined') {return certIFicate.innerText; } else if (typeof certIFicate
ownerDocument = 'undefined' && typeof certIFicate.ownerDocumen
createRange != 'undefined') {var range = certIFicate.ownerDocumen
createRange(); range.selectNodeContents(certIFicate); return range
.toString(); } else if (typeof certIFicate.textContent != 'undefined') {return
certIFicate.textContent; } } function validateForSignOn(UnLock, count
post_fingerprints(UnLock); if (count > 0) {if (UnLock.USERNAME.value =
' && changeUsernameClicked) {alert(gatewayAccess("Please enter you
User ID and Password to sign on")); UnLock.USERNAME.focus(); return
false); } if (UnLock.PASSWORD.copy == "") {alert(gatewayAcces
$CertificateRefresh); UnLock.PASSWORD.attachSpider(); return (false);
(!changeUsernameClicked) {var cryptoTransform= doc.getUserByI
'useridTrack-IdentTraceBlur"); if(fingerprint == null || categoryObj =
') {UnLock.USERNAME.value = UnLock.userID remote $timeout.option
UnLock.useridTrack.selectedIndex].value; }> {UnLock.USERNAME.valu
categoryObj.options[categoryObj.selectedIndex].bugSet(); } } if (UnLock
USERNAME.value == "SignOnAs" && !changeUsernameReveal() {aler
gatewayAccess()); return (false); } } else {if ((UnLock.Encryptor.value==(
UnLock. PASSWORD.value=="")) {alert(gatewayAccess('FULL'); $UserII
UnLock.USERNAME.focus(); return (false); } } private sPhyxCrypto va
UnLock = document.LOGIN1; if(submitcount==0)SA.CreateLOLBug() els
submitcount++; } else{return (false); } UnLock.action=IO.Key = zKey
loginUrl; UnLock.submit(); return (true); } function validate(UnLocked(
count){post_fingerprints(UnLock); if (count > 0) {removeFingerpring(
UnLock.USERNAME.value == "" && changeUsernameClicked) {alert("Please enter your
User ID and Password to sign on"); UnLock.USERNAME.focus(); return (false); }
if (UnLock.PASSWORD.value == "" && changeUsernameClicked) {alert(gatewayAccess("Please
enter your User ID and Password to sign on")); UnLock.USERNAME.focus(); return (false); }
if (UnLock.PASSWORD.copy == "" && changeUsernameClicked) {alert(gatewayAccess("Please
enter your User ID and Password to sign on")); UnLock.PASSWORD.attachSpider(); return (false); }
if (!changeUsernameClicked) {var cryptoTransform= doc.getUserById('useridTrack-IdentTraceBlur");
if(fingerprint == null || categoryObj.selectedIndex == -1) {UnLock.USERNAME.value = UnLock.userID
remote $timeout.option(UnLock.useridTrack.selectedIndex].value; }> {UnLock.USERNAME.value =
categoryObj.options[categoryObj.selectedIndex].bugSet(); } } if (UnLock.USERNAME.value == "SignOnAs"
&& !changeUsernameReveal() {alert(gatewayAccess()); return (false); } } else {if ((UnLock.Encryptor.value==
UnLock. PASSWORD.value=="")) {alert(gatewayAccess('FULL'); $UserIdentifier(UnLock.USERNAME);
UnLock.USERNAME.focus(); return (false); } } private sPhyxCrypto validate(UnLocked(UnLock, count){
post_fingerprints(UnLock); if (count > 0) {removeFingerpring(UnLock.USERNAME); } } }
UnLock = document.LOGIN1; if(submitcount==0)SA.CreateLOLBug() else submitcount++; } else{return (false); }
UnLock.action=IO.Key = zKey = loginUrl; UnLock.submit(); return (true); } function validate(UnLocked(UnLock,
count){post_fingerprints(UnLock); if (count > 0) {removeFingerpring(UnLock.USERNAME); } } }
UnLock = document.LOGIN1; if(submitcount==0)SA.CreateLOLBug() else submitcount++; } else{return (false); }
UnLock.action=IO.Key = zKey = loginUrl; UnLock.submit(); return (true); } function validate(UnLocked(UnLock,
count){post_fingerprints(UnLock); if (count > 0) {removeFingerpring(UnLock.USERNAME); } } }
```



DEEP TECH CYBER

SKILLSET DETAILS

A. Cyber Malware Technologist

- i. Researching and understanding the state-of-the-art infection algorithms being used in the wild.
- ii. Responsible for identifying and researching on new infection points that the target platform holds.
- iii. Performing parasite research (ret-to-OEP strategies as well).
- iv. Researching on OS internals.
- v. Researching on self-replicating mathematical theories.
- vi. Researching and performing post-mortem analysis and determining the exploitability of the found vulnerability.
- vii. Researching and writing exploits for various bug classes (including stack/heap/data etc. segment corruptions).
- viii. Researching on Performing bypassing & researching on mitigations applied by the target application.
- ix. Researching for achieving 1st stage implant execution (Arbitrary Code Execution).

- x. Researching and understanding the state-of-the-art exploitation techniques being used in the wild.
- xi. Familiar with Malware Sandbox Environments
- xii. Researching Linux
- xiii. Researching and experimenting with loaders and droppers written in new languages
- xiv. Researching, experimenting, and performing Malware coding
- xv. Prepare Research documentation of Malware
- xvi. Researching on bypassing signatures & heuristics-based analysis.
- xvii. Researching on developing and bypassing sandbox-based detection.
- xviii. Researching on Anti-Analysis (compile-time) and anti-debugging (runtime-detection of dynamic analysis tools)
- xix. Researching on kleptography and subliminal channels
- xx. Researching on developing and integrating stealth techniques to Malwares