TOP 10 PRIVILEGE ESCALATION HACKS FOR THE MAINFRAME

"These files are secure, you would need special and operations privileges to access them!"

I've heard this statement from countless mainframe security professionals who have spent their career managing mainframe External Security Managers (ESM) like IBM's Resource Access Control Facility (RACF) or Broadcom's Top Secret and Access Control Facility (ACF2). These ESMs are extremely effective at assigning permissions to specific accounts so only appropriate users can access each dataset. It gives mainframe security professionals peace of mind that their data is protected and safe. Unfortunately, this sense of security is false: it ignores the fact that mainframe hackers have uncovered several extremely effective ways to escalate privileges! This means they can start with a normal, restricted user and upgrade to a privileged account with special and operations permissions without the mainframe security manager ever knowing. Once they have those privileged permissions, they have full control over the mainframe and the highly sensitive data it touches.

"I'm not worried about privilege escalation because you would already need access to a mainframe account and mainframes are unhackable!"

This is another common sentiment amongst the mainframe community that ignores the very real risk to their systems. I've helped teams identify their mainframe attack surface that hackers will explore and the top six ways that mainframe penetration testers have found success with gaining an initial foothold on the mainframe[1]. Even worse, these attacks don't include the increasingly common problem of insider threats, which up to 59 percent of organizations experienced over the last 12 months[2]. Insider threats already have valid credentials and permissions on the mainframe and are one privilege escalation attack away from having the power to exfiltrate sensitive data, install ransomware, or significantly impact the operational capacity of the mainframe. If you have a mainframe in your IT infrastructure, then you are already aware of the devasting business implications of an attack like this.

So how are attackers able to bypass the restrictions that system programmers put in place? Below are the top 10 ways a simple setting can be used to provide a restricted user with complete control over the mainframe:

1. Access to any Authorized Program Facility (APF) authorized libraries can be leveraged to gain special and operations privileges. RACF will store your credentials' permissions in memory, which will be referenced anytime you attempt to access a resource. Through an APF authorized library, you can use an open source hacker tool called elv.apf[3] to automatically modify the section of memory to change your permissions to anything you desire.
2. If you have a user account that does not have access to an APF authorized library, maybe you can try to create one. Check to see if your account has the BPX.FILEATTR.APF setting. If you do, you can use the *extattr +a* command to change anything to an APF authorized library and use step 1 for special and operations permissions.
3. If you can't access or create an APF authorized library directly, try to use a data concatenation path attack to reach a previously denied dataset. Thanks to nested permissions, if you have higher level access to a dataset, and there is an APF authorized library beneath it, then you can trick RACF into thinking you have appropriate permissions and can use it much like the methods above.
4. Similar to the data concatenation path attack, if your account has DASDVOL permissions, you can copy and move an APF authorized library into a path that is executable for your account, which again provides you the ability to execute the first step to escalate privileges.
5. If your account doesn't have the ability above, then check to see if you have access to an account that does. IBM® z/OS® regularly uses surrogate privileges so that an account can submit a job or script as another account. If you have surrogate privileges for an account that can do any of the steps above, then you can use that surrogate account to give yourself special and operations permissions.
6. Another often forgotten permission is BPX.SUPERUSER which enables the account to have superuser, or root, privileges on USS which can then be leveraged to access or modify any files, including APF authorized libraries.
7. Another extremely common way for penetration testers to gain special and operations permissions on a production machine is through the Network Job Entry (NJE) service. Pentesters will often first exploit a development machine, where the above permissions are far less likely to be protected. Once they have the initial foothold on a mainframe that is a trusted node, they will use NJE to submit a job to every production machine simultaneously. This, which can create them a unique privileged account on each logical partition or automatically submit privileged jobs and scripts. Even scarier, the NJE password is stored in JES2PARMLIB and can be spoofed for a man-in-the-middle attack to simulate a trusted node.
8. One of the easiest ways to escalate privileges is to get a password for a privileged account. Security hygiene has improved tremendously over the past two decades, but production mainframes tend to be set up and untouched to avoid any potential issues that can result in costly downtime. Professional pentesters regularly remark that they are able to do quick

searches for password datasets and files on z/OS or z/Linux to find a list of credentials that they can use to escalate privileges. These files are often forgotten about until a hacker uncovers and uses them for nefarious purposes.

9. The RACF PassTicket is a one-time-only password that is generated by a requesting product or function.[4] If your account has access to this feature you can generate a single-use password for a privileged account and directly log in with their credentials for full control of the mainframe.
10. All the above revolve around permissions and misconfigurations that are extremely common, despite the efforts of highly competent system programmers, because the dynamic environment and tremendous amount of user accounts are nearly impossible to manage perfectly. Yet, even if you have a perfect security posture where none of the above succeed, you can still be vulnerable to a 0Day software exploit. There are very few security researchers looking for software exploits for the mainframe, which gives many system programmers the belief that they are safe. Unfortunately, it is much more likely that this leaves them more vulnerable. Pirate Boy co-founder Gottfrid Svartholm developed several 0Day attacks while attacking the Swedish Nordea Bank to create more than 20 RACF special user IDs.[5] You also have companies like Key Resources Inc. who have a Vulnerability Analysis Program (VAP) solution that scans software for potential memory and permission violations and often finds dozens of potential exploits in the most common software products on the market[6]. A quick Google search of IBM® Guardium® on the Common Vulnerability and Exposures database will return over 40 vulnerabilities in that solution alone.[7]

When you address the significant challenges in effectively defending from privilege escalation attacks, it becomes clear that mainframe security professionals require assistance in the immediate identification of users who are able to find a way to escalate their privileges. BMC AMI Security's endpoint detection and response solution keeps track of all users' permissions and will immediately notify the Security Operations Center and mainframe administrators, in real-time, when a user's credentials are modified and they are able to access previously denied datasets. This enables the security team to immediately respond and remediate before a potential breach or ransomware devastates the valuable data on the mainframe. If you'd like to know more about this software solution, please contact Christopher_perry@bmc.com or visit BMC Software for more information.

[1] https://www.dev.blogs.bmc.com/top-6-ways-a-hacker-will-gain-access-to-your-mainframe/

[2] https://www.helpnetsecurity.com/2019/04/05/detect-insider-threats/

[3] https://github.com/ayoul3/privesc

[4] https://www.ibm.com/support/knowledgecenter/en/SSLTBW_2.3.0/com.ibm.zos.v2r3.icha700/pass.htm

[5] https://www.share.org/blog/the-mainframe-security-threat,-inside-and-out

[6] https://www.krisecurity.com/vulnerability-analysis

[7] https://www.cvedetails.com/vulnerability-list/vendor_id-14/product_id-32588/IBM-Security-Guardium.html