We ran some vulnerability scanning on open ports on Drexel domain. As seen in the image below there were a number of vulnerabilities found on that open IP.

A screenshot of a computer

Description automatically generated with medium confidence

In the mixed vulnerabilities folder, we located some high and critical vulnerabilities

A screenshot of a computer

Description automatically generated with medium confidence

After looking up the high vulnerability we found out that, a security bypass vulnerability exists on the remote SSH server owing to a fault in the keyboard-interactive authentication procedures. The auth2-chall.c method kbdint\_next\_device() incorrectly limits the processing of keyboard-interactive devices to a single connection. A remote attacker can utilize a forged keyboard-interactive 'devices' string to get around the regular 6-login-attempt limit (MaxAuthTries), allowing them to execute a brute-force attack or trigger a denial-of-service issue.

We search Shodan for vulnerabilities in database

Graphical user interface, text, application, email

Description automatically generated

One Vulnerability that was found was CVE-2018-15919

According to NIST, when the target user existed, the OpenSSH server responded differently to unsuccessful GSSAPI authentication attempts than when the target user did not exist. This flaw might be used by a remote attacker to check for the existence of certain usernames on a target machine.

We ran a Nessus scan on the same IP address which was found on Shodan. We tried to play around with of the settings. And for the assessment we made it custom so it can look for the database as well.

Graphical user interface, text, application

Description automatically generated

Below in the image is our scan results which showed mostly INFO about Vulnerabilities

A screenshot of a computer

Description automatically generated with medium confidence

After opening one of the info, we found that an ICMP timestamp request is answered by the remote host. This gives an attacker access to the date set on the target system, which might let an unauthenticated, remote attacker overcome time-based authentication methods. Timestamps supplied by Windows Vista / 7 / 2008 / 2008 R2 PCs are intentionally wrong but are frequently within 1000 seconds of the real system time.

This could be resolved easily by filtering out the ICMP timestamp requests (13), and the outgoing ICMP timestamp replies (14).