Exploit Payloads a11y.text Exploit Payloads Working with Exploit Payloads a11y.text Working with Exploit Payloads Metasploit helps deliver our exploitÂ payloads against aÂ target system. When creating an Exploit Payload, we have severalÂ thingsÂ to consider, from the operating system architecture, to anti-virus, IDS, IPS, etc. In evading detection of our exploits, we will want to encode our payloads to remove any bad characters and add some randomness to the final output using NOPs. Metasploit comes with a number of payload encoders and NOP generators to help aid us in this area. Select a payload encoder : Must not touch certain registers Must be under the max size Must avoid BadChars Encoders are ranked Select a nop generator : Tries the most random one first NOPs are also ranked Payload Encoding Example a11y.text Payload Encoding Example The defined Payload Space is 900 bytes The Payload is 300 bytes long The Encoder stub adds another 40 bytes to the payload The NOPs will then fill in the remaining 560 bytes bringing the final payload.encoded size to 900 bytes The NOP padding can be avoided by adding â€˜DisableNopsâ€™ => true to the exploit Payload Block Options a11y.text Payload Block Options As is the case for most things in the Framework, payloads can be tweaked by exploits. â€˜StackAdjustmentâ€™ prefixes â€œsub espâ€• code â€˜MinNopsâ€™, â€˜MaxNopsâ€™, â€˜DisableNopsâ€™ â€˜Prefixâ€™ places data before the payload â€˜PrefixEncoderâ€™ places it before the stub These options can also go into the Targets block, allowing for different BadChars for targets and allows Targets to hit different OS architectures. Next MSFvenom Prev Exploit Targets