Relatório de testes

Ambiente testado: http://sandbox.meuspedidos.com.br:8080/
Features testados: Login
Período de testes: 02/10/2015 a 04/10/2015
Tester: Angelo Luis Ribeiro das Neves

1. Cenários testados

- Testes manuais
  - Comportamentos do formulário de login
    - Com dados reais
    - Com dados inexistentes
    - Com dados inválidos
    - Em diferentes browsers
    - Em diferentes sistemas operacionais
  - Segurança
    - Ataques de força bruta
    - Ataques de DoS
    - Vazamento de dados via log
- Testes automatizados
  - Comportamentos do formulário de login
    - Com dados reais
    - Com dados inexistentes
    - Com dados inválidos

2. Resultados

2.1. Falhas críticas

2.1.1. Ataques de força bruta

Durante a realização dos testes manuais foram observados 3 fatores que geraram uma suspeita de que o sistema é vulnerável a ataques de força bruta:

- O sistema apresenta mensagens de erro diferentes quando um email digitado está cadastrado e quando não está cadastrado, mesmo com a senha errada nos dois casos.

- O sistema permite inúmeras tentativas seguidas de login para a mesma conta com a senha errada, sem nenhum mecanismo de proteção
- O sistema permite o uso de senhas extremamente simples como por exemplo 4 digitos. (10.000 combinações possíveis)

Com base nestes comportamentos torna-se possível programar uma automatização que realiza um ataque de dicionário no campo e-mail (ex: [nomeComum]@[principaisDominios].com.br) até encontrar um e-mail cadastrado.

Uma vez encontrado um email podemos iniciar ataques no campo senha até encontrar uma combinação que realize o login com sucesso. É razoável imaginar que alguns usuários utilizam senhas simples com apenas 4 digitos, ao limitar o range de senhas a este caso podemos otimizar o processo.

Para testar este conceito foi desenvolvido um script simples em Python que parte de um email existente e aplica força bruta no campo senha com combinações de 4 digitos.

O script monta uma lista de senhas possíveis e divide a senha entre vários processos que realizam as requisições simultâneamente, com esta técnica foi possível encontrar a senha correta em poucos minutos. O Script está disponível juntamente com o projeto de testes no caminho "brute_force/terminal.py"

2.1.2. Ataques DoS

O script de força bruta utilizado no teste descrito no item 2.1.1 também pode ser utilizado para testes de sobrecarga do sistema através do formulário de login. Basicamente o script inicia um determinado número de processos que realizam uma série de tentativas de login no site.

Quanto maior o número de processos entre os quais o set de senhas possíveis será distríbuido menor é o tempo necessário para encontrar a senha e maior é a carga aplicada no servidor. Conforme a quantidade de processos foi ampliada nos testes notou-se uma crescente instabilidade no servidor. Ao executar um teste com 10.000 senhas distribuídas entre 20 processos simultâneos o servidor parou de responder após alguns minutos.

A queda de serviço ocorreu na noite do dia 03/10/2015 e foi verificada através de diferentes conexões de internet, comprovando que o servidor realmente estava fora do ar. O serviço foi parcialmente restaurado na noite do dia 04/10/2015 quando a página de login voltou para o ar porém ao tentar acessar o sistema ainda ocorre um erro 500. O Script está disponível juntamente com o projeto de testes no caminho "brute_force/terminal.py" e o log dos testes mencionados estão no arquivo "brute_force/DoS_server_down.log"

2.1.3. Vazamento de dados via log

Conforme mencionado no item 2.1.2 o servidor voltou ao ar no dia 04/10/2015 após falhas decorrentes de um teste de DoS, porém ainda apresentando erro 500 ao tentar utilizar o sistema. Notou-se que as telas de erro estão configuradas de forma a deixar transparecer diversas informações a respeito do servidor e do sistema devido ao uso do parâmetro DEBUG = True nas configurações do Django. Alguns dos dados expostos:

- **Framework:** Django 1.3.7
- **Contas de administração:** 'Dev', 'dev@meuspedidos.com.br'
- **Webserver:** gunicorn/0.13.4
- **Linguagem:** Python 2.7.3
- **Banco de dados:** 'ENGINE': 'django.db.backends.mysql',
    - 'HOST': '127.0.0.1',
    - 'NAME': 'representante',
    - 'PORT': '3306',
    - 'TIME_ZONE': 'America/Sao_Paulo',
    - 'USER': 'root'

Em posse de informações como estas é possível iniciar uma pesquisa de exploits conhecidas para os sistemas em uso no servidor, bem como tentar obter mais informações através de outros erros. Em último caso pode ser possível assumir o controle da aplicação ou do servidor. Como este tipo de teste não está contemplado no escopo solicitado a situação não foi investigada a fundo. O log completo de dados vazados está no arquivo "brute_force/log_website_error.pdf"

2.2. Falhas não críticas

2.2.1. Comportamento da validação de email

Foi constatado que o sistema apresenta um comportamento irregular na validação do campo email. Foram implementadas duas camadas de validação, uma via HTML5 que impede o envio do formulário e outra no backend do sistema caso um e-mail inválido seja enviado.

A implementação da validação HTML5 não está padronizada entre os browsers, de forma que um mesmo endereço de email gera um erro de browser em alguns casos e um erro de sistema em outros, dependendo do browser utilizado e do sistema operacional.

Neste caso a validação HTML 5 poderia ser removida para garantir uma experiência padronizada, porém como em ambos os casos o email inválido não é aceito, considera-se que este problema não é crítico e os testes automatizados

criados foram preparados para aceitar qualquer um dos dois tipos de validação como retorno aceitável para emails inválidos

2.3. Resultados conformes

2.3.1. Comportamento do formulário de login

Excetuando-se o caso mencionado no item 2.2.1, todos os comportamentos do formulário de login se mostraram conformes tanto nos testes manuais quanto nos automatizados, em diversos navegadores e sistemas operacionais incluindo mobile. Pontualmente hoje dia 04/10/2015 o primeiro teste automatizado (usuário e senha válidos) está falhando pois o sistema está apresentando erro 500 ao tentar autenticar o usuário, o que indica que o teste está adequado.

3. Sugestões e conclusão

Primeiramente cabe informar que em virtude das falhas no ambiente de testes após a realização do teste de DoS descrito no item 2.1.2 o desenvolvimento dos testes automatizados foi interrompido prematuramente, mesmo assim considera-se que abrangem os cenários mais comuns.

Conforme demonstrado no início deste documento, foram encontradas vulnerabilidades de segurança no sistema e por isso é possível listar 3 sugestões iniciais para minimizar de imediato a exposição a ameaças:

- Implementar um sistema de proteção contra força bruta na tela de login (ex: Captcha)
- Implementar um sistema de proteção contra DoS (ex: bloqueio de IPs por excesso de requisições, CloudFlare, etc)
- Desabilitar a exibição de logs de erro detalhados em um ambiente que está publicamente disponível, principalmente caso este apresente as mesmas características gerais do ambiente de produção.

Por fim gostaria de agradecer pela oportunidade de participar deste processo seletivo. Espero que de alguma forma estas informações possam contribuir para a melhoria do produto da Meus Pedidos e espero poder ter o privilégio de me juntar a esta equipe.