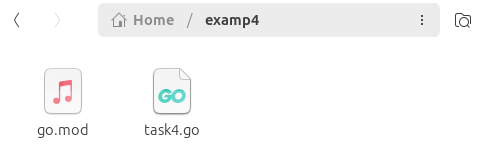
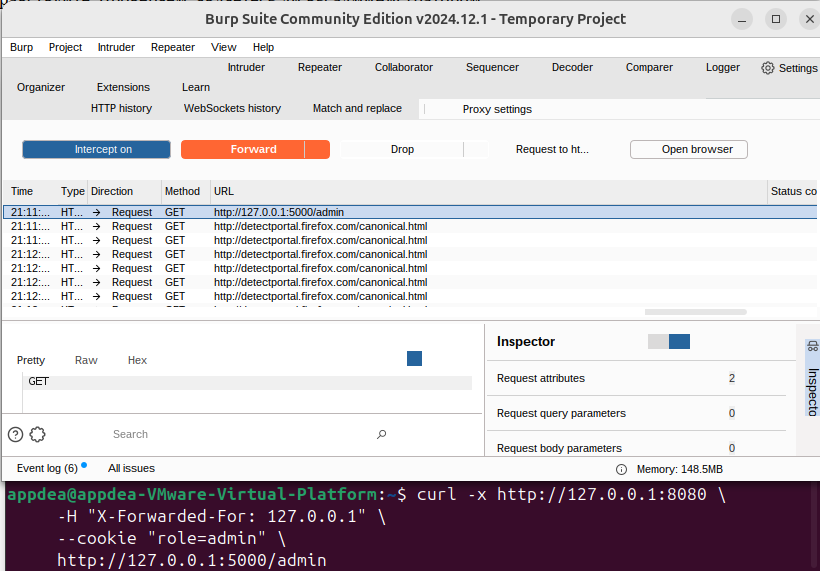
1. Собираем проект. Используем go run task4.go, инициализировав go.mod перед этим. Нужно для наглядной эксплуатации





1. Для запуска корректируем код, чтобы обращаться к куку role, иначе это вызывает «панику», когда пытаемся получить h.Role.Value.

*if h.Role != nil && strings.ToLower(h.Role.Value) == "admin" {*

*// Проверка IP и отображение админки*

*}*

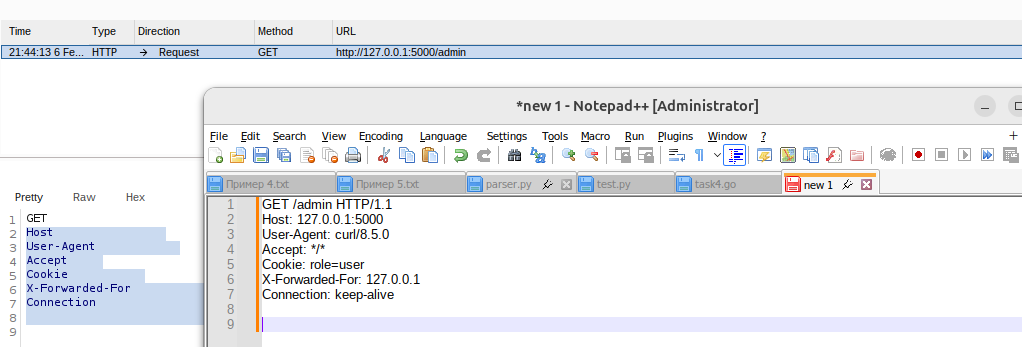
1. Основная проблема в заголовке X-Forwarded-For для проверки Ip-адреса. Это ненадёжно и это мы проэксплуатируем.
2. Также отсутствует какой-либо механизм аутентификации и авторизации.

**КАК ВОСПРОИЗВЕСТИ:**

1. Как и говорилось в начале, запускаем сервер на порту 5000.
2. Через курл отправляем запрос и ловим его при помощи BurpSuite

*curl -x http://127.0.0.1:8080 -H "X-Forwarded-For: 127.0.0.1" -b "role=user" http://127.0.0.1:5000/admin*

1. К сожалению, графическая оболочка, запущенная через JAR работает крайне печально, но перехватить пакет мы всё-таки смогли

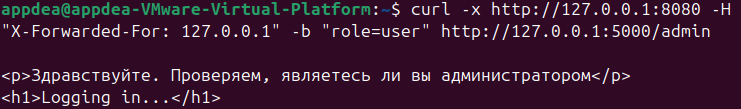


1. Дело за малым, меняем содержимое запроса следующим образом:

Заголовок: X-Forwarded-For: 127.0.0.1 → X-Forwarded-For: localhost

Куки: role=user → role=admin

1. Результатом успешной эксплуатации является следующее сообщение:



**Как защитить:**

1. Один из вариантов – использовать список доверенных адресов (например, доверенных прокси), у которых будет доступ к админке.

*if h.Role != nil && strings.ToLower(h.Role.Value) == "admin" {*

*// Список доверенных IP-адресов*

*trustedIPs := []string{"127.0.0.1", "192.168.1.1"} // Добавьте сюда доверенные IP*

*for \_, trustedIP := range trustedIPs {*

*if trustedIP == h.ClientIP {*

*fmt.Fprintln(w, html\_AdminDashboard())*

*return*

*}*

*}*

*}*

1. Постараемся получить реальный IP – адрес, используя функцию getClientIP, в которой будет использован r.RemoteAddr:

*func getClientIP(r \*http.Request) string {*

*// Если сервер находится за доверенным прокси, используем X-Forwarded-For*

*forwarded := r.Header.Get("X-Forwarded-For")*

*if forwarded != "" {*

*// Берем первый IP из списка (клиентский IP)*

*ips := strings.Split(forwarded, ", ")*

*if len(ips) > 0 {*

*return ips[0]*

*}*

*}*

*// Если X-Forwarded-For отсутствует, используем RemoteAddr*

*ip, \_, err := net.SplitHostPort(r.RemoteAddr)*

*if err != nil {*

*return r.RemoteAddr*

*}*

*return ip*

*}*

1. Дополнительными мерами могут служить использование HTTPS, внедрение механизма аутентификации и авторизации и т.д