Limitare il traffico in ingresso con nginx

La gestione del traffico in entrata è una delle funzionalità essenziali per garantire la sicurezza e le prestazioni di un server web. nginx, grazie alla sua architettura efficiente e flessibile, offre vari metodi per limitare e controllare il traffico. In questo articolo, esploreremo come utilizzare nginx per limitare il traffico in entrata, migliorando così la sicurezza e le prestazioni del server.

nginx è un server web e proxy inverso molto popolare, noto per la sua capacità di gestire un gran numero di connessioni simultanee con un basso utilizzo delle risorse. Oltre alle sue funzionalità di base, nginx offre strumenti avanzati per il controllo del traffico, che possono essere utilizzati per prevenire attacchi DDoS, ridurre il carico del server e migliorare l'esperienza degli utenti.

Uno dei metodi principali per limitare il traffico è controllare la velocità di connessione. Questo può essere fatto utilizzando il modulo `ngx_http_limit_conn_module` di nginx.

Aggiungi la seguente configurazione nel blocco `http` del file `nginx.conf`:

```
http {
    limit_conn_zone $binary_remote_addr
zone=addr:10m;
    server {
        location / {
            limit_conn addr 10;
            # Altre direttive
        }
    }
}
```

```
}  
}
```

In questa configurazione:

- `limit_conn_zone $binary_remote_addr zone=addr:10m;` definisce una zona condivisa di 10 MB per tracciare le connessioni.
- `limit_conn addr 10;` limita il numero di connessioni simultanee per indirizzo IP a 10.

Oltre a limitare il numero di connessioni, puoi anche controllare la quantità di dati che ogni client può scaricare in un determinato periodo di tempo. Questo è possibile utilizzando il modulo `ngx_http_limit_req_module`.

Aggiungi la seguente configurazione nel blocco `http` del file `nginx.conf`:

```
http {  
    limit_req_zone $binary_remote_addr  
zone=req_limit:10m rate=1r/s;  
    server {  
        location / {  
            limit_req zone=req_limit burst=5 nodelay;  
            # Altre direttive  
        }  
    }  
}
```

In questa configurazione:

- `limit_req_zone $binary_remote_addr zone=req_limit:10m rate=1r/s;` crea una zona di 10 MB per

tracciare le richieste e imposta un limite di 1 richiesta al secondo per indirizzo IP.

- `limit_req zone=req_limit burst=5 nodelay;` permette una "esplosione" di 5 richieste oltre il limite configurato senza ritardi.

Per bloccare gli indirizzi IP sospetti o noti per attività dannose, puoi utilizzare il modulo `ngx_http_access_module`.

Aggiungi la seguente configurazione nel blocco `server` o `location` del file di configurazione del tuo sito:

```
server {
    location / {
        deny 192.168.1.1;
        allow all;
        # Altre direttive
    }
}
```

In questa configurazione:

- `deny 192.168.1.1;` blocca l'indirizzo IP specificato.
- `allow all;` permette l'accesso a tutti gli altri indirizzi IP.

Se stai eseguendo un'API, limitare la frequenza delle richieste può prevenire l'abuso delle risorse. Configura il rate limiting nel file di configurazione del tuo server:

```
http {
    limit_req_zone $binary_remote_addr
zone=api_limit:10m rate=10r/m;
    server {
```

```
        location /api/ {
            limit_req zone=api_limit burst=20
nodelay;
            # Altre direttive
        }
    }
}
```

In questa configurazione:

- `limit_req_zone $binary_remote_addr zone=api_limit:10m rate=10r/m;` imposta un limite di 10 richieste al minuto per indirizzo IP.
- `limit_req zone=api_limit burst=20 nodelay;` permette una "esplosione" di 20 richieste oltre il limite configurato senza ritardi.

Conclusione

Limitare il traffico in entrata con nginx è un metodo efficace per migliorare la sicurezza e le prestazioni del tuo server. Utilizzando i moduli integrati di nginx, puoi facilmente configurare limitazioni di connessione, limitazioni di banda e blocco degli IP maliziosi. Ricorda di monitorare regolarmente il traffico e adattare le configurazioni in base alle esigenze specifiche del tuo ambiente.