		arduo il lavoro dell'attaccante abilitandone l'identificazione, come i client che applicano la crittografia o il login prima di procedere con il vero lavoro (ovviamente ciò non vuol dire che gli accessi non debbano essere crittografati).
Risorse di sistema	Sfruttare le funzionalità messe a disposizione dal Sistema Operativo	Utilizzare le impostazioni del sistema operativo.

Tabella 17 - STRIDE: Indirizzamento del Denial of Service

Saturazione della rete. Se si dispone di strutture statiche di connessione, cosa accade se queste si saturano? L'utilizzo di firewall può fornire un livello di protezione basato su ACLs per controllare l'accettazione (o l'invio) di dati e possono essere utili per mitigare gli attacchi di negazione di servizio di rete.
Individuazione delle risorse esauribili. Si possono identificare tre tipologie distinte di risorse esauribili: la prima è quella relativa alle risorse di rete; la seconda è quella relativa a quelle risorse direttamente gestite lato codice; la terza è quella relativa alle risorse gestite dal sistema operativo. In ogni caso, le risorse flessibili (elastic resources) risultano sempre essere una tecnica preziosa. Ad esempio, negli anni 90 alcuni stack TCP avevano un limite hardcoded di cinque connessioni TCP semiaperte (una connessione semiaperta è una connessione che viene attivata nel processo di avvio. Non bisogna preoccuparsi del fatto che ciò potrebbe non avere un senso, piuttosto bisognerebbe chiedersi il motivo per cui questo limite fu impostato a cinque). Oggi è spesso possibile ottenere risorse flessibili dai vari tipi di cloud provider.
Risorse di sistema. I sistemi operativi tendono ad avere limiti o quote per controllare il consumo di risorse a livello di codice applicativo. Considerare le risorse gestite dal sistema operativo, come la memoria o l'utilizzo del disco. Se il codice applicativo viene eseguito su server dedicato, può essere ragionevole consentirgli di utilizzare tutte le risorse di cui quel server dispone. Prestare attenzione a porre gli opportuni limiti al codice e assicurarsi di documentare quanto prestabilito.
Risorse del programma. Acquisire consapevolezza circa le limitazioni che si potrebbero avere nella gestione delle risorse applicative. Ad esempio, un attaccante potrebbe portare l'applicazione software ad un dispendio di lavoro e risorse inviando un flusso dati che richiederebbe costose operazioni crittografiche che potrebbero esporre il software stesso a vulnerabilità di "Denial Of service".
La seguente tabella riporta le vulnerabilità della Top 10 OWASP 2017[27] riconducibili alle minacce di denial of service, e per ciascuna vulnerabilità indicata, le relative pratiche[28] e requisiti[29] di sicurezza consigliati da OWASP:

OWASP TOP-10 2017 **(Rischi di sicurezza delle applicazioni)**	**OWASP Proactive Controls 2018 v 3.0** **(Pratiche di sicurezza proattive)**	**OWASP ASVS 3.0** **(Requisiti di sicurezza applicative)**
A3 - Sensitive Data Exposure	C10 - Handle All Errors and Exceptions	V8 - Error Handling and Logging
A6 - Security Misconfiguration	C3 - Secure Database Access	V19 - Configuration

[27] https://www.owasp.org/index.php/Category:OWASP_Top_Ten_2017_Project

[28] https://www.owasp.org/index.php/OWASP_Proactive_Controls

[29] https://github.com/OWASP/ASVS