

Cybersecurity course [Application Security]

Emilio Coppa

coppa@diag.uniroma1.it

Sapienza University of Rome



whoami



- Assistant Professor, DIAG, Sapienza University of Rome
- Research interests: program analysis, vulnerability detection, reverse engineering, malware analysis
- National organizer of CyberChallenge.IT in 2017-2019
- Local organizer at Sapienza of CyberChallenge.IT
- Mail: **coppa@diag.uniroma1.it**
- Office hours: *send me an email to choose a slot*

Cybersecurity course (9 CFU)

Lower levels security
Prof. Fabrizio d'Amore
6 CFU

Application level security
Prof. Emilio Coppa
3 CFU

Website: <https://sites.google.com/diag.uniroma1.it/cybersecurity/>

Topics

Email security	<ul style="list-style-type: none">- Background: SMTP, architecture, POP/IMAP, message fields, MIME, ESMTP- Security problems: confidentiality, integrity, spam, spoofing, (spear) phishing- SPF- DKIM- DMARC- ARC- S/MIME and OpenPGP
Web technologies	<ul style="list-style-type: none">- HTTP, HTTPS, HTTP3+QUIC, cookies, local storage, web socket, DOM, RESP API, JWT, SPA, Authentication (Basic, digest, OAuth/SSO)
Web security: attacks and defenses	<ul style="list-style-type: none">- The cost of vulnerability- OWASP: principles, top categories- Server-side risks and their mitigations: path traversal, command-line injection, SQL injection, others (directory permissions, directory listing, symbolic links, daemon privilege), WAF- Client-side risks and their mitigations: session hijacking, XSS, CSRF, SOP, Cookie attacks, CSP- Hybrid risks: MITM, SSL Strip Attack, HSTS, CORS
Anonymity	<ul style="list-style-type: none">- Cookie tracking and profiling- TOR: architecture, routing, services

Lectures (expected...)

week	month	day		
1	october			
		wednesday	5	09.00-11.00
		friday	7	10.00-12.00
friday		21	10.00-12.00	
2		friday	28	10.00-12.00
3	november	friday	4	10.00-12.00
4		friday	11	10.00-12.00
5		friday	18	10.00-12.00
6		friday	25	10.00-12.00
7	december	friday	2	10.00-12.00
8		wednesday	7	09.00-11.00
9		wednesday	14	09.00-11.00
10		friday	16	08.00-12.00
11		wednesday	21	09.00-11.00
		friday	23	10.00-12.00

Check out our Google Calendar!



[\[CLICK HERE TO ADD IT TO YOUR CALENDAR\]](#)

Slides and other resources

Join Piazza to get access to the material using your institutional email (if you do not have one, please send me an email): piazza.com/uniroma1.it/fall2022/10600393

Sapienza-University of Rome - Fall 2022

10600393: Cybersecurity - Web Security part

Add Syllabus

Course Information

Staff

Resources

Edit Resource Sections

General Information

Manually sort using

Sort on:

General Information		Actions
Official course website	≡	<div><div>Edit</div><div>Post a note</div><div>Update link</div><div>Delete</div></div>
Course syllabus	≡	<div><div>Edit</div><div>Post a note</div><div>Update link</div><div>Delete</div></div>
Log of lectures for the web security part	≡	<div><div>Edit</div><div>Post a note</div><div>Update link</div><div>Delete</div></div>

Requirements

- Basic understanding: UDP, TCP, HTTP, SMTP, client/server paradigm
- Able to write a few basic scripts
- Able to quickly learn what it is needed about HTML, CSS, Javascript, PHP
- Basic knowledge of some cryptographic primitives: you will learn in detail some of these during the lectures of Prof. d'Amore.
- **Eager to learn by playing with things!**

Books

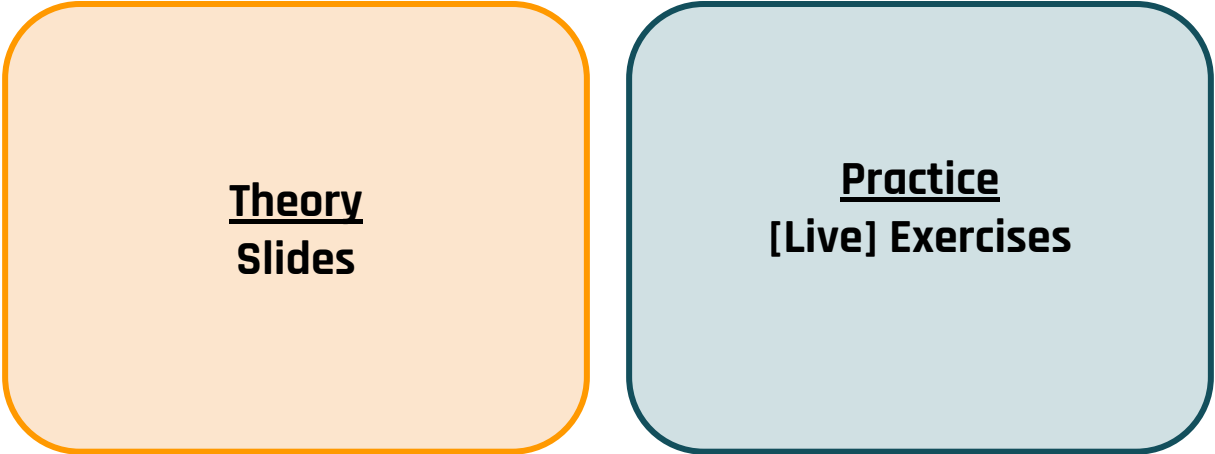
Email security:

- I could not find any “easy to study” book on the most advanced topics that we will see in the lectures. See the reference in the slides for good online resources.

Web security:

- **Web Application Security. A. Hoffman. O'Reilly. 2020.** Easy to read, not very detailed. Good for grasping the main ideas.
- **Real-World Bug Hunting: A Field Guide to Web Hacking. Peter Yaworksi. No Starch Press. 2019.** Nice book with many examples from real-world vulnerabilities.
- **The Web Application Hacker's Handbook. D. Stuttard and M. Pinto. Wiley. 2011.** More detailed than the first one.
- **The Tangled Web. Michal Zalewski. No starch press. 2012.** Different but very interesting book. It explains how the web (in 2012) worked.

How lectures will be organized



The diagram consists of two rounded rectangular boxes side-by-side. The left box is light orange with an orange border and contains the text 'Theory Slides'. The right box is light blue with a dark blue border and contains the text 'Practice [Live] Exercises'. Both boxes have rounded corners.

**Theory
Slides**

**Practice
[Live] Exercises**

Exam (for this part of the course!)

Written Exam

Theory

1.5 CFU

0-15 points

Practical Exam

Practice

1.5 CFU

0-15 points

Written exam

- 5 exam sessions during the year
- It will be done immediately after the written exam for the other part of course
- It will last 30 minutes
- It will contain 2 *macro* questions
- Very easy if you attend the course and check the slides

Practical exam

There are three **alternatives**:

- A. Play a CTF during the course (see later):
 - pros: learn by playing, just do a few web “exercises”
 - cons: you can only do the CTF during the course

- B. Help me prepare new challenges for the CTF (see later):
 - pros: very easy if you have experience with the topics and CTFs
 - cons: available only to selected students, must be done in the first 1.5 months of the course

- C. Study in depth two vulnerabilities from real-world applications (see later)
 - pros: you can do at any time during the year
 - cons: real-world code can be tricky to understand

Practical Exam - Plan A: Play the CTF

Practical exam

Capture-The-Flag (CTF) Competition!

WebHackIT 2022

Wait, what is a CTF?

A **Capture the Flag (CTF)** is a computer security competition. CTF contests are usually designed to serve as an educational exercise to give participants hands-on experience in the sort of attacks and protections found in the real world. There are two main styles of capture the flag competitions: jeopardy - like this competition - and attack/defense.

Jeopardy-style competitions usually involve **multiple categories of problems, each of which contains a variety of questions that range from easy to more difficult ones**. Individuals attempt to earn the most points in the competition's time frame, **but do not directly attack each other**. CTFs often touch many aspects of information security, like cryptography, steganography, binary analysis and exploitation, reverse engineering, web security, mobile security and others. **This competition will be focused on email security and web security (client and server-side).**

[Home](#) / [CTFs](#) / [Events](#) / [Upcoming](#)<https://ctftime.org/>

CTF Events

All

Upcoming

Archive

Format ▾

Location ▾

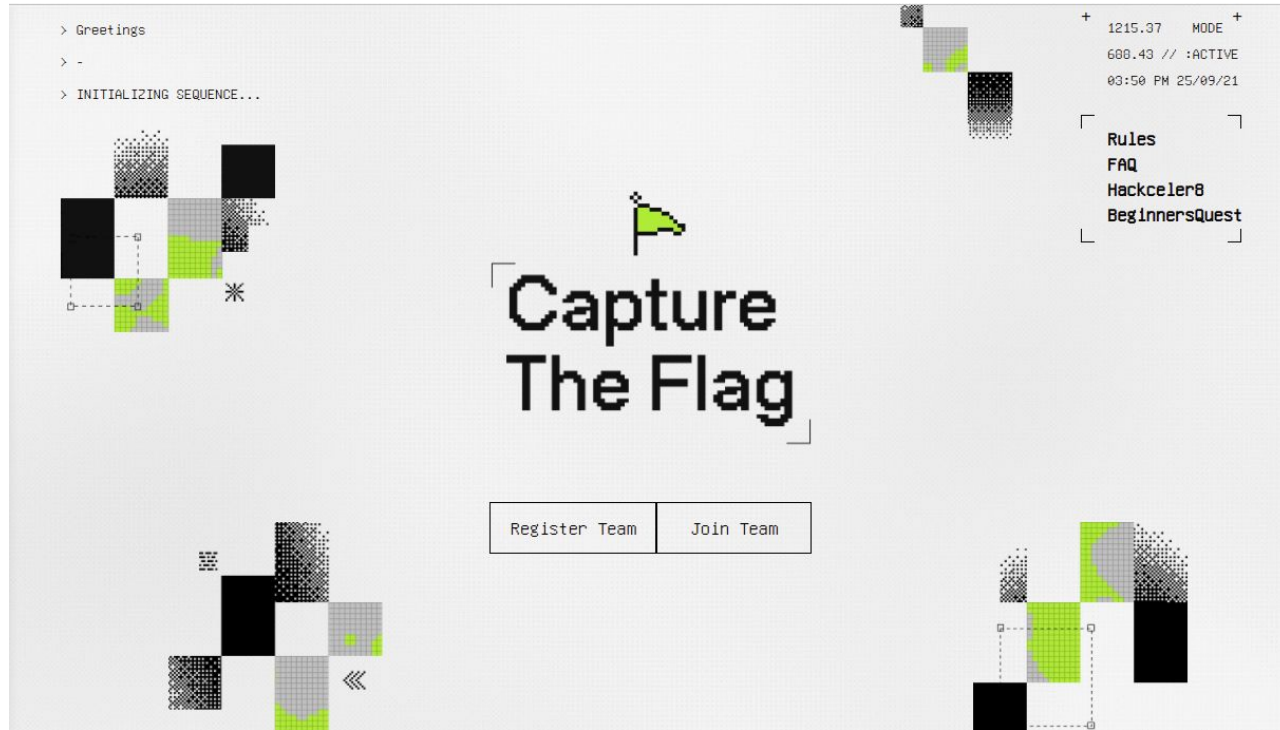
Restrictions ▾

2021



Name	Date	Format	Location	Weight	Notes
DeconstruCT.F 2021	01 Ott., 08:07 UTC — 02 Ott. 2021, 08:07 UTC	Jeopardy	On-line	0,00	26 teams will participate
TastelessCTF 2021	01 Ott., 18:00 UTC — 03 Ott. 2021, 18:00 UTC	Jeopardy	On-line	35,64	42 teams will participate
TSG CTF 2021	02 Ott., 07:00 UTC — 03 Ott. 2021, 07:00 UTC	Jeopardy	On-line	34,80	30 teams will participate
Sacramentum	02 Ott., 11:30 UTC — 03 Ott. 2021, 06:30 UTC	Jeopardy	On-line	0,00	2 teams will participate
RuCTF 2021	03 Ott., 07:00 UTC — 03 Ott. 2021, 15:00 UTC	Attack-Defense	On-line	0,00	33 teams will participate
pbctf 2021	09 Ott., 00:00 UTC — 11 Ott. 2021, 00:00 UTC	Jeopardy	On-line	24,90	13 teams will participate
DamCTF 2021	09 Ott., 00:00 UTC — 11 Ott. 2021, 00:00 UTC	Jeopardy	On-line	24,43	12 teams will participate

Big IT companies: Google CTF...



Facebook CTF...



University: iCTF (UC Santa Barbara - G. Vigna)

The iCTF evolved from a number of security “live exercises” that were carried out locally by Prof. [Giovanni Vigna](#) at UC Santa Barbara, in 2001 and 2002.

Motivated by the student’s enthusiasm for security competitions, Prof. Vigna carried out the first wide-area edition of the iCTF in December 2003. In that CTF, fourteen teams from around the United States competed in a contest to compromise other teams’ network services while trying to protect their own services from attacks. This historical contest included teams from UC Santa Barbara, North Carolina State University, the Naval Postgraduate School in Monterey, the West Point Academy, Georgia Tech, University of Texas at Austin, and University of Illinois, Urbana-Champaign.

In 2004, the iCTF evolved into a truly *international* exercise (hence, the name “iCTF”), which included teams from the United States, Austria, Germany, Italy, and Norway.

For many years, the iCTF was the world’s largest educational security competition, and helped popularizing this type of event.

Throughout the years, new competition designs have been introduced that innovated the more “traditional” designs followed in the early editions of the competition.

More precisely, in 2008 the iCTF featured a separate virtual network for each team. The goal was to attack a terrorist network and defuse a bomb after compromising a number of hosts. This competition allowed for the recording of several parallel multi-stage attacks against the same network. The resulting dataset has been used as the basis for correlation and attack prediction research.

In 2009, the participants had to compromise the browsers of a large group of simulated users, steal their money, and create a botnet. This design focused particularly on the concept of drive-by attacks, in which users are lured into visiting web sites that deliver attacks silently.

DEF CON CTF...

`./ 000 --- DEF CON CTF`



DEF CON CTF 2021 QUALS

Quals were held online, 48 hours, starting at UTC 00:00 May 1st (CTFtime).

Congrats PPP! Final announcement

We have also released most challenges to our archive, and challenge authors are starting to release their code on github.

The scoreboard (including solves information) is archived at scoreboard2021.ooverflow.io

Itching to know what other players tried and how they reached the flag? PCAPs are available for {baby mama gran}-a-fallen-lap-ray, cozen (~280 GB), exploit-for-dummies, mooosl, mra, nooombers, nooopster, pooow-{buddy pal}, qoo-or-ooo, back-to-qoo, rick, signalooo, threefactoorx, tiamat.



ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/cose

Computers
&
Security

TC 11 Briefing Papers



Cybersecurity knowledge and skills taught in capture the flag challenges



Valdemar Švábenský*, Pavel Čeleda, Jan Vykopal, Silvia Brišáková

Masaryk University, Brno, Czech Republic

Capture the Flag as Cyber Security Introduction

Lucas McDaniel
University of Alaska Fairbanks
lamcdaniel@alaska.edu

Erik Talvi
University of Alaska Fairbanks
etalvi@alaska.edu

Brian Hay
University of Alaska Fairbanks
brian.hay@alaska.edu

Abstract

Introducing technical concepts to students with little to no technical background can be a challenging task for any teacher to achieve. The concept of gamification has been introduced recently as a method to motivate students by taking a variety of techniques found in popular games, and adding them into educational modules. Extending from this notion, it has been found that capture the flag (CTF) style competitions are a successful way to introduce students to a variety of technical concepts within the standard computer science curriculum.

memorizing answers or shortcuts to get answers to bypass the lectures and materials that they perceive as boring [9]. This means that they will not only fail to fully grasp the topic at hand, but will quickly forget it after the course or even the lecture has finished.

Therefore many instructors have moved to using hands-on learning techniques so that students are able to explore the application of security concepts in a real world environment. In the past this has been done with labs and tutorials including those that are instructor-guided or others that are exploratory in nature, but recently there has been a trend to gamify the environment.

Earn points by solving challenges

Each challenge:

- It is a web application (in WebHackIT!)
- It is designed to reproduce a real-world problem inside an isolated environment
- It may simplify some assumptions and details to help understand the goal
- In some cases, you have to show knowledge of some topics... they are just exercise
- While in other cases, you have to find and then exploit a vulnerability!

How?

- The solution will depend on the specific challenge
- During the course, I will show you how to solve some challenges
- ...some of these challenges have been taken from other past CTFs
- I will show you some tools that can help you

You think it will like this...



It will be like this:



How do you show that you have solved the challenge?

- Each challenge contains a secret string (which is hard to guess!)
- This is called a **flag**... hence the name Capture-The-Flag!
- A flag could be obtained after successfully performing some “tasks” or it is the result of attack (e.g., you can login as an admin)
- Easy to recognize since the format will be: **WIT{<random chars>}** [hard to guess!]
- Just insert the the flag on the web portal to get the points!

How many challenges?

- ~13 training challenges: we will solve them together!
- ~22 challenges: different complexity levels
 - beginner
 - easy
 - almost-medium

If you have a bit of experience with web challenges in other CTFs, then it will be VERY easy for you... even boring (hence, there is PLAN B)!

When?

- ~13 training challenges:
 - **during the course!**
- ~22 challenges:
 - **three weeks (+1 week in the case of problems) towards the end of the course**
 - [planned] from November 18 to December 16
 - you can solve the challenge at any time during this timeframe!

How to get access to the challenges?

A web portal will be soon be available at:

<https://webhack.it/>

You need to register to get an account!

WebHackIT 2022

WebHackIT 2022 is the Capture The Flag hacking contest of the **Cybersecurity course** at Sapienza University of Rome. The competition is organised by **Emilio Coppa**.

Wait, what is a CTF?

A **Capture the Flag (CTF)** is a computer security competition. CTF contests are usually designed to serve as an educational exercise to give participants hands-on experience in the sort of attacks and protections found in the real world. There are two main styles of capture the flag competitions: jeopardy - like this competition - and attack/defense.

Jeopardy-style competitions usually involve multiple categories of problems, each of which contains a variety of questions that range from easy to more difficult ones. Teams or individuals attempt to earn the most points in the competition's time frame, but do not directly attack each other. CTFs often touch many aspects of information security, like cryptography, steganography, binary analysis and exploitation, reverse engineering, web security, mobile security and others. This competition will be focused on email security and web security (client and server-side).

How the score is assigned?

- This is an experiment! Hard to say now, but...
- If you solve the training challenges and most beginner/easy challenges then you will get easily 11 points out of 15 points!
- The exact list of challenges needed for 11 points will be announced before the CTF
- For the remaining 4 points, I will see the ranking during the CTF to assign them.

Do we have to solve ALL challenges to get the MAX score?

No, a CTF is different from an exam. Some challenges may be broken and unsolvable. Some challenges may require a bit of guessing. Some challenges may be boring.

I do not expect you to solve all the challenges. If most of you are solving ALL the challenges, then you are likely cheating (more later).

How the points are assigned for each challenge?

Dynamic score:

$$\# \text{ points} = \text{floor}(500 / X^{**0.15})$$

where X is the number of solvers and the score is updated for all solvers after each solve.

This means:

- $X = 1 \Rightarrow 500$ points
- $X = 10 \Rightarrow 353$ points
- $X = 100 \Rightarrow 250$ points
- ...

If a challenge is solved by few students, then it means it was harder and then they will get more points than another challenge that was solved by hundreds of students.

Can we solve the challenges in groups?

No, the competition is played individually by each student.

What if we cheat and we just share the flags?

Three reasons why you should not cheat:

1. Ethical reasons

- It is not fair to your colleagues that are not cheating
- It is not fair to your professor:

I am putting a lot of effort in this competition! I am trying to make you learn cybersecurity by playing a game! Please, keep this in your mind!

What if we cheat by just sharing the flags?

Three reasons why you should not cheat:

2. **If you share a flag, you are decreasing your points**

The dynamic score mechanism will reduce the number of points assigned for a challenge based on the number of players that have solved it.

What if we cheat and we just share the flags?

Three reasons why you should not cheat:

3. **After submitting a flag, you have to also submit a write-up**

A **write-up** is a report that explains in detail and step-by-step the challenge. If you do not submit a write-up for a challenge, you will not get the score for the exam. I will check them carefully when assigning your score.

YOU have to submit flags that YOU have obtained by solving the challenges. This is true also for the training challenges (where I show you how to solve them)!

What if you cheat....

If I have a suspect that you have cheated, I will do not say anything until the day you pass the written exam and then I communicate to you that your score for the practical part of the exam is invalid.

What if you cannot play the CTF?

The CTF lasts three weeks to allow anybody to find the time to play it. However, if you cannot play in that time frame, you can look at PLAN C.

What is it allowed in the CTF?

There are still some ground rules during a CTF:

- No attacks against the CTF platform that is keeping track of the points
- No disruptive attacks against the challenges, including denial-of-service (DoS), floods, DNS poisoning, ARP spoofing, MITM, etc...
- Don't try to bruteforce flags by doing an unreasonable number of attempts in the flag submission page.
- Vulnerability/network scanners are mostly useless and not allowed. Do not use any automatic tool!
- If you find a way to hack into the platform, do not exploit it but report it to the professor. I will give you extra points!

When in doubt, ask the professor.

Practical Exam - Plan B: Help me with the CTF

Preconditions

- You need to be familiar with the topics of the course
- You have experience with CTFs (i.e., you have played them in the past)
- You have (a bit of) experience with docker
- You are available in the first 1.5 months of the course

What you have to do

- Pick two vulnerability types
- Propose one challenge for each vulnerability type
- Discuss with me about your proposals
- Develop the challenges

I may decide to deploy your challenges during the CTF as “extra challenges”.

Practical Exam - Plan C: Analysis of vulnerabilities

What you have to do

- Choose one popular open-source web application/framework, e.g., [WordPress](#), [Joomla](#), [Django](#), Flask, etc.
- Look at the CVEs / bugs / vulnerabilities reported by the community for the application. E.g., Google it: "<application> CVE" to find a list of CVEs.
- Select 1-2 vulnerabilities: they should be related to the topics of the course. Do not choose naive / shallow bugs. One vulnerability is enough if you pick something that is interesting to discuss. **Write to me to let me know which vulnerabilities you would like to consider: I will check them and give you an OK/KO.**

What you have to do (2)

- Write a report (max 10 pages) that:
 - briefly describe the architecture of the web application / framework
 - describe the type(s) of vulnerability that you have considered
 - describe the vulnerabilities in detail
 - describe how to exploit the vulnerabilities
 - describe how the vulnerabilities were patched by the community
 - (needed only for getting max points) describe (and provide with the report) a docker environment that is able to reproduce the vulnerabilities

There is no strict deadline. You can do the written exam even before submitting the project.

DISCLAIMER

In this course, we will see some attack techniques, which may even work in the real world.

Keep in mind that you are not allowed to perform the same attacks in the real world:

**YOU MAY GET IN LEGAL PROBLEMS IF YOU DO NOT HAVE THE PERMISSION TO PERFORM AN
ATTACK ON A PLATFORM/WEBSITE. THE REAL LIFE IS NOT A GAME!
HOWEVER, YOU SHOULD SEE THIS AS A POSSIBLE JOB OPPORTUNITY!**

Il delitto di accesso abusivo ad un sistema informatico

L'art.615-ter, va considerato, unitamente al 640 ter, l'articolo più importante introdotto dalla legge 547 del 1993 poiché rende penalmente perseguibile l'accesso abusivo ad un sistema informatico telematico protetto da misure di sicurezza o il mantenimento in esso contro la volontà espressa tacita dell'avente diritto

Tale articolo recita testualmente:

Art.615-ter. *(Accesso abusivo ad un sistema informatico e telematico).*

Chiunque abusivamente si introduce in un sistema informatico o telematico protetto da misure di sicurezza ovvero vi si mantiene contro la volontà espressa o tacita di chi ha il diritto di escluderlo, è punito con la reclusione fino a tre anni.

La pena è della reclusione da uno a cinque anni:

- 1. se il fatto è commesso da un pubblico ufficiale o da un incaricato di un pubblico servizio, o con abuso dei poteri o con violazione dei doveri inerenti alla funzione o al servizio, o da chi esercita anche abusivamente la professione di investigatore privato, o con abuso della qualità di operatore del sistema;*
- 2. se il colpevole per commettere il fatto usa violenza sulle cose o alle persone, ovvero se è palesemente armato;*
- 3. se dal fatto deriva la distruzione o il danneggiamento del sistema o l'interruzione totale o parziale del suo funzionamento, ovvero la distruzione o il danneggiamento dei dati, delle informazioni o dei programmi in esso contenuti.*

Bug bounty programs

If you do find -- in some legal ways -- a vulnerability in a real-world (web) application, check if the owner has a bug bounty program.

However, be cautious when contacting the owner since requesting for a bug bounty -- when a program is not in place -- could be seen as an illegal request.