Studio Project
ALPR

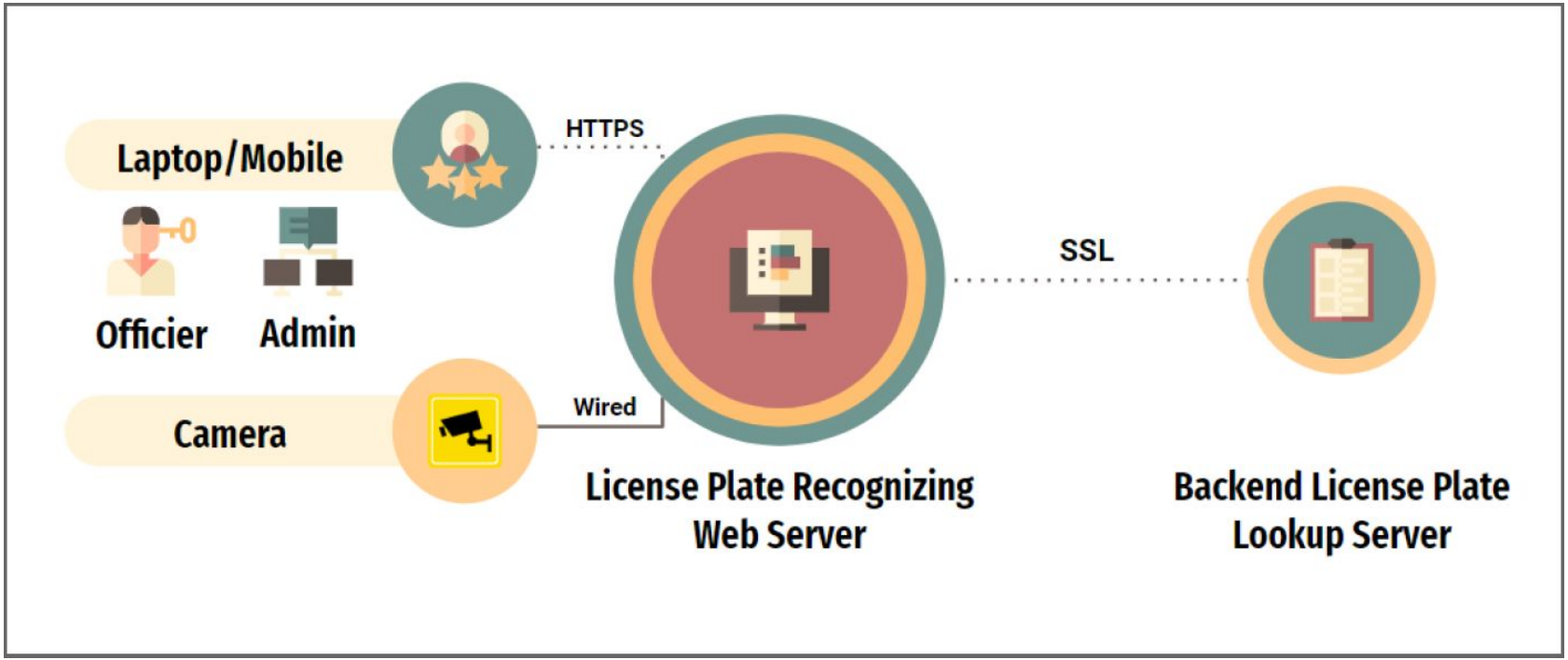
The Team 3 (Purple)

Project Members and Roles

- Developer Artifacts and Documentation / Vulnerability Assessment
  - Youngmi Choi (최영미), Seungkyu Lee (이승규)
- Client Design and Implementation / Reverse Engineering
  - Seongsik Kim (김성식), Haenggi Lee (이행기)
- Server Design and Implementation / Penetration Testing
  - Kibong Song (송기봉), Gyunggui Moon (문경귀)
- Mentor / Support
  - David Belasco (데이비드)

Evaluation Target - The Team 2 (Ahnlab)

- Their work was solid and the documentation was informative.



Executive Summary

- Found at least 3 vulnerabilities that could compromise the assets that Team 2 was trying to protect (8 vulnerabilities found in total)

Priority	Vulnerability	Impact	Recommendation
High (CVSS 8.7)	If we can set request.session.error, then we have access to add a new user in the same privilege as super user.	An attacker can compromise user credential information.	Fix improper error handling; Apply least privileged access control
High (CVSS 7.5)	The key value used for DB encryption in the Lookup server is the ASCII string '2Team_AhnLab'; it's hard-coded and predictable.	An attacker can disclose personal identifiable information.	Remove the hard-coded key value; Change to a stronger one and store it in a secure place
High (CVSS 7.9)	0 can be entered as the maximum user setting value in 'server.conf'. This causes the server to either return an internal error or hang.	An attacker can tamper configuration information to stop the service.	Check setting input value whether it is within the normal range before use

Evaluation Narrative

- Focus on critical/high threats specified in the Team 2 threat modeling
- Make our selections based on our interests rather than prioritizing them
- Admit we tend to look for low-hanging fruit from source code

Analysis Technique	**Rationale**	**Vulnerability**
Code Review ⭕	The source code tells the detailed implementation.	VID-1, VID-4, VID-5
Dumb Fuzzing ❌	Users can input images into the OpenALPR library.	None
Input Validation ⭕	Users can upload (any) files to the client app.	VID-2, VID-3, VID-8
Use Exploit Tools ⭕	The client app uses popular OSS.	VID-6
Research Vulnerability DB ❌	The client app uses a specific version of OSS.	None
Tinker with the System ⭕	Keep Calm and Carry On	VID-7

Evaluation Resource

- Try to use all the practice learned in the course

Analysis Technique	**Resource**	**Difficulty**
Code Review ⭕	Patience, Coffee	High
Dumb Fuzzing ❌	zzuf, JPG fuzzer Result: Caught in OpenCV assertions upon both fuzzing	High
Input Validation ⭕	ImageMagick, Browser	Mid
Use Exploit Tools ⭕	OWASP ZAP, Metasploitable in Kali	High
Research Vulnerability DB ❌	https://www.exploit-db.com/ https://www.cvedetails.com/ Result: Known vuls are already patched to Django 4.0.5 and opencv-python 4.6.0.66 (They are latest)	Low
Tinker with the System ⭕	Wireshark, db_dump, Hex Viewer, IDA, Browser	Mid

Vulnerabilities found by Code Review

Vulnerability	Location	POC
VID-1: If we can set request.session.error, then have access to add a new user in the same privilege as super user. High (CVSS 8.7) Elevation-of-Privilege	client\...\alpr\views.py > upload() video_path = settings.MEDIA_ROOT + '/..' + document.uploadedFile.url pic = Image.open(video_path) width, height = pic.size if (width*height <= 1): request.session['error'] = True	1) Make a small image with width and height less than or equal to 1 2) Upload it 3) Then we have access to add a new user in the same privilege as super user.
VID-4: The key value used for DB encryption is the ASCII string '2Team_AhnLab'; it's hard-coded. High (CVSS 7.5) Information Disclosure	server\...\server.cpp const char code[12] = { 0x32, 0x54, 0x65, 0x61, 0x6d, 0x5f, 0x41, 0x68, 0x6e, 0x4c, 0x61, 0x62 }; ... ret = dbp->set_encrypt(dbp, code, flags);	1) db_dump -P 2Team_Ahnlab licensplate.db > dump.txt 2) Inspect dump.txt with Hex Viewer 3) Then we can see decrypted data in DB. (see next slide)

```
86  else if ( (*(unsigned int (__fastcall **)(__int64, int *))(v57 + 824))(v57, v66)
87          || (v66[0] = 1, (*(unsigned int (__fastcall **)(__int64, const char *, __int64))(v57
88                                                                                           v57,
89                                                                                           code,
90                                                                                           116)) )
91  {
92    printf("DB Encrypt Error\n");
93    result = 0xFFFFFFFFi64;
94  }
```

```
.rdata:000000014000DB30 ; const char code[12]
.rdata:000000014000DB30 code            db '2', 'T', 'e', 'a', 'm', '_', 'A', 'h', 'n', 'L', 'a'
.rdata:000000014000DB30                                         ; DATA XREF: ProcessClient(void *)+B9↑o
.rdata:000000014000DB30                 db 'b'
.rdata:000000014000DB3C                 align 20h
```

HxD - [제목없음2]

파일(F) 편집(E) 찾기(S) 보기(V) 분석(A) 기타 설정(X) 창 설정(W) ?

16 | ANSI | 16 진수

제목없음1 | test2.txt | 제목없음2

```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00004420  6F 72 65 72 20 53 70 6F 72 74 0A 74 65 61 6C 00  orer Sport.teal.
00004430  47 58 56 32 39 34 31 00 47 58 56 32 39 34 31 0A  GXV2941.GXV2941.
00004440  4F 77 6E 65 72 20 57 61 6E 74 65 64 0A 30 31 2F  Owner Wanted.01/
00004450  30 36 2F 32 30 32 33 0A 43 68 72 69 73 74 6F 70  06/2023.Christop
00004460  68 65 72 20 46 69 73 68 65 72 0A 30 36 2F 31 31  her Fisher.06/11
00004470  2F 31 39 38 31 0A 39 30 32 39 20 47 6C 6F 72 69  /1981.9029 Glori
00004480  61 20 53 74 72 61 76 65 6E 75 65 0A 45 61 73 74  a Stravenue.East
00004490  20 41 6E 64 72 65 77 2C 20 4D 44 20 36 37 33 35   Andrew, MD 6735
000044A0  39 0A 32 30 30 33 0A 43 61 64 69 6C 6C 61 63 0A  9.2003.Cadillac.
000044B0  45 6E 63 6F 72 65 0A 62 6C 61 63 6B 00 4B 36 36  Encore.black.K66
000044C0  33 31 39 4B 00 4B 36 36 33 31 39 4B 0A 4F 77 6E  319K.K66319K.Own
000044D0  65 72 20 57 61 6E 74 65 64 0A 30 34 2F 32 38 2F  er Wanted.04/28/
000044E0  32 30 32 34 0A 52 75 73 73 65 6C 6C 20 4A 6F 68  2024.Russell Joh
000044F0  6E 73 0A 31 30 2F 31 35 2F 31 39 35 31 0A 33 39  ns.10/15/1951.39
00004500  32 20 42 61 72 62 61 72 61 20 48 65 69 67 68 74  2 Barbara Height
00004510  73 0A 4A 6F 68 6E 6E 79 66 75 72 74 2C 20 41 4C  s.Johnnyfurt, AL
00004520  20 38 33 39 30 39 0A 32 30 31 36 0A 42 4D 57 0A   83909.2016.BMW.
00004530  53 36 0A 61 71 75 61 00 4C 42 56 36 31 35 37 00  S6.aqua.LBV6157.
00004540  4C 42 56 36 31 35 37 0A 4E 6F 20 57 61 6E 74 73  LBV6157.No Wants
00004550  20 2F 20 57 61 72 72 61 6E 74 73 0A 31 30 2F 32   / Warrants.10/2
00004560  33 2F 32 30 32 32 0A 4E 69 63 68 6F 6C 61 73 20  3/2022.Nicholas
00004570  57 69 6C 73 6F 6E 0A 30 31 2F 30 31 2F 31 39 36  Wilson.01/01/196
00004580  32 0A 36 30 31 32 20 42 72 61 6E 64 6F 6E 20 50  2.6012 Brandon P
00004590  6F 72 74 73 0A 4E 6F 72 74 68 20 44 61 76 69 64  orts.North David
000045A0  62 6F 72 6F 75 67 68 2C 20 4F 4B 20 31 36 31 38  borough, OK 1618
000045B0  39 0A 31 39 39 39 0A 50 6C 79 6D 6F 75 74 68 0A  9.1999.Plymouth.
000045C0  32 35 30 30 20 52 65 67 75 6C 61 72 20 43 61 62  2500 Regular Cab
000045D0  0A 74 65 61 6C 00 4C 42 58 39 30 35 31 00 4C 42  .teal.LBX9051.LB
000045E0  58 39 30 35 31 0A 4E 6F 20 57 61 6E 74 73 20 2F  X9051.No Wants /
000045F0  20 57 61 72 72 61 6E 74 73 0A 30 36 2F 30 31 2F   Warrants.06/01/
00004600  32 30 32 33 0A 4B 69 6D 62 65 72 6C 79 20 4D 61  2023.Kimberly Ma
00004610  79 6E 61 72 64 0A 30 32 2F 32 34 2F 31 39 37 31  ynard.02/24/1971
00004620  0A 39 33 34 32 20 4C 6F 72 69 20 42 79 70 61 73  .9342 Lori Bypas
00004630  73 20 53 75 69 74 65 20 37 31 31 0A 45 61 73 74  s Suite 711.East
00004640  20 53 61 6E 64 72 61 2C 20 43 4F 20 39 32 31 36   Sandra, CO 9216
00004650  32 0A 32 30 31 39 0A 4C 69 6E 63 6F 6C 6E 0A 4D  2.2019.Lincoln.M
00004660  6F 6E 74 65 72 6F 20 53 70 6F 72 74 0A 77 68 69  ontero Sport.whi
```

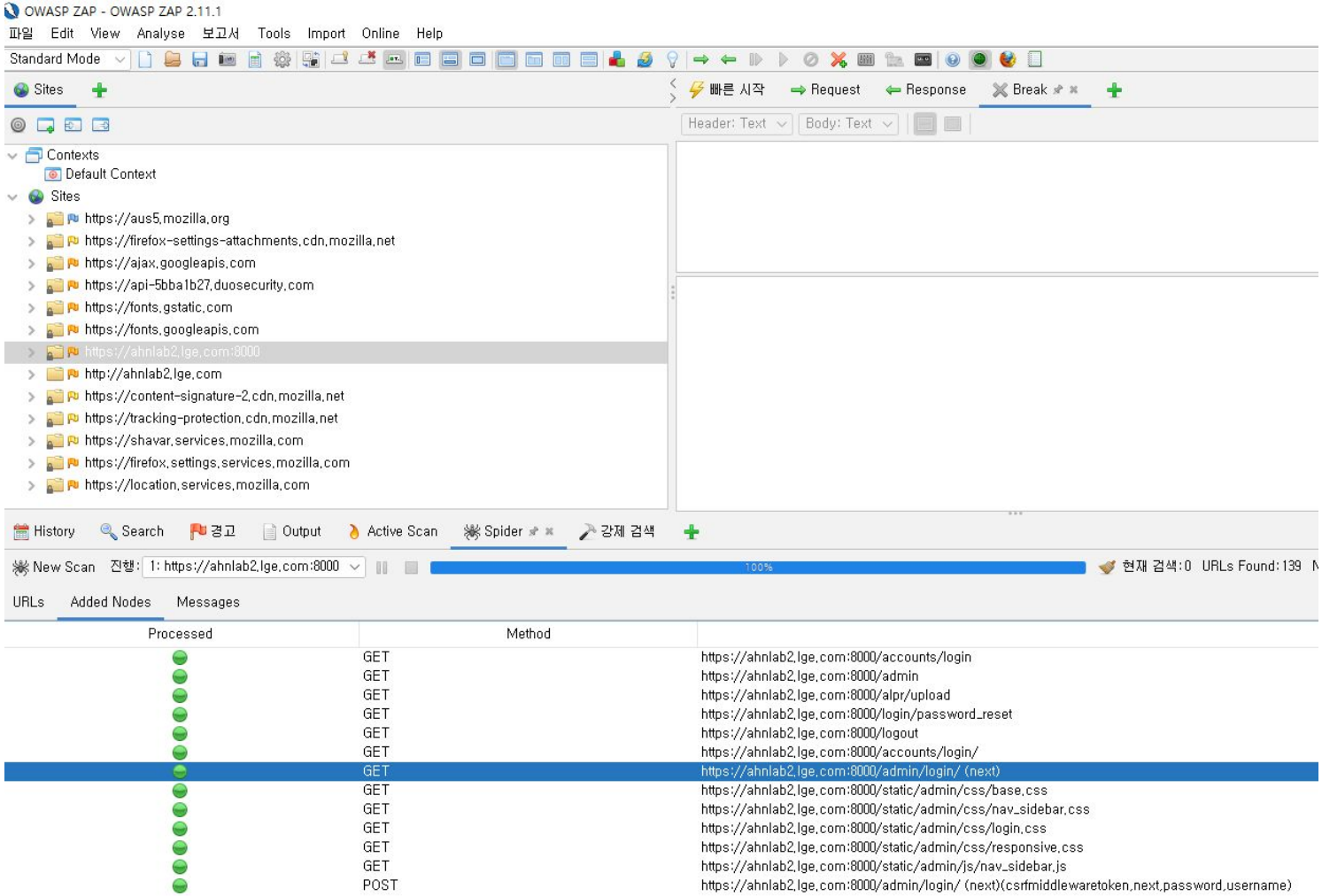
Vulnerability found by Input Validation

Vulnerability	Location	POC

VID-3: If you rename the sample video file to a long name and upload it, an internal error occurs the moment the license plate is recognized.
 	
High (CVSS 7.3)
 	
Denial-of-Service
	client\...\alpr\views.py > VideoUpstream > __del__
	
	`def __del__(self):`
	`    if self.video:`
	`        self.video.release()`
	`        print("video released")`
	`    print("destroy............")`
	
	`def get_frame(self, frame=None):`
	`    try:`
	`        if frame is None:`
	`            image = self.frame`
	`            _, jpeg = cv2.imencode('.jpg', image)`
	`        else :`
	`            image = frame`
	`            _, jpeg = cv2.imencode('.jpg', image)`
	`    except:`
	`        print("close connection")`
		1) Rename beaver1.avi to aa.avi
		2) Upload it
		3) update error: Storage can not find an available filename for "media/gord36k/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa_OK0dS1s\261_099cwwv.jpg". Please make sure that the corresponding file field allows sufficient "max_length".
		video released
		destroy............

Vulnerability found by Exploit Tool

Vulnerability	Location	POC
VID-6: It is possible to delete files uploaded by another user. High (CVSS 7.6) Tampering	client\...\alpr\views.py > remove() id = request.GET['id'] filepath = settings.BASE_DIR url = models.Document.objects.get(id=id).uploadedFile.url try: models.Document.objects.filter(id=id).delete() print(rootpath+'../'+url) os.remove(rootpath+'../'+url)	1) Select one file 'id' (e.g. 54) in the DB 'alpr_document' table 2) Login to the client 3) Type https://ahnlab2.lge.com:8000/alpr/remove?id=54 in the browser 4) If the file deletion is successful, nothing is displayed. Otherwise, a 500 server error returns. 5) Check if the file 'id' has been removed in the DB 'alpr_document' table


OWASP ZAP - OWASP ZAP 2.11.1
파일 Edit View Analyse 보고서 Tools Import Online Help
Standard Mode
Sites
빠른 시작 Request Response Break
Header: Text Body: Text
Contexts
Default Context
Sites
https://aus5.mozilla.org
https://firefox-settings-attachments.cdn.mozilla.net
https://ajax.googleapis.com
https://api-5bba1b27.duosecurity.com
https://fonts.gstatic.com
https://fonts.googleapis.com
https://ahnlab2.lge.com:8000
http://ahnlab2.lge.com
https://content-signature-2.cdn.mozilla.net
https://tracking-protection.cdn.mozilla.net
https://shavar.services.mozilla.com
https://firefox.settings.services.mozilla.com
https://location.services.mozilla.com
History Search 경고 Output Active Scan Spider 강제 검색
New Scan 진행: 1: https://ahnlab2.lge.com:8000
100%
현재 검색: 0 URLs Found: 139
URLs Added Nodes Messages
Processed Method
GET https://ahnlab2.lge.com:8000/accounts/login
GET https://ahnlab2.lge.com:8000/admin
GET https://ahnlab2.lge.com:8000/alpr/upload
GET https://ahnlab2.lge.com:8000/login/password_reset
GET https://ahnlab2.lge.com:8000/logout
GET https://ahnlab2.lge.com:8000/accounts/login/
GET https://ahnlab2.lge.com:8000/admin/login/ (next)
GET https://ahnlab2.lge.com:8000/static/admin/css/base.css
GET https://ahnlab2.lge.com:8000/static/admin/css/nav_sidebar.css
GET https://ahnlab2.lge.com:8000/static/admin/css/login.css
GET https://ahnlab2.lge.com:8000/static/admin/css/responsive.css
GET https://ahnlab2.lge.com:8000/static/admin/js/nav_sidebar.js
POST https://ahnlab2.lge.com:8000/admin/login/ (next)(csrfmiddlewaretoken,next,password,username)

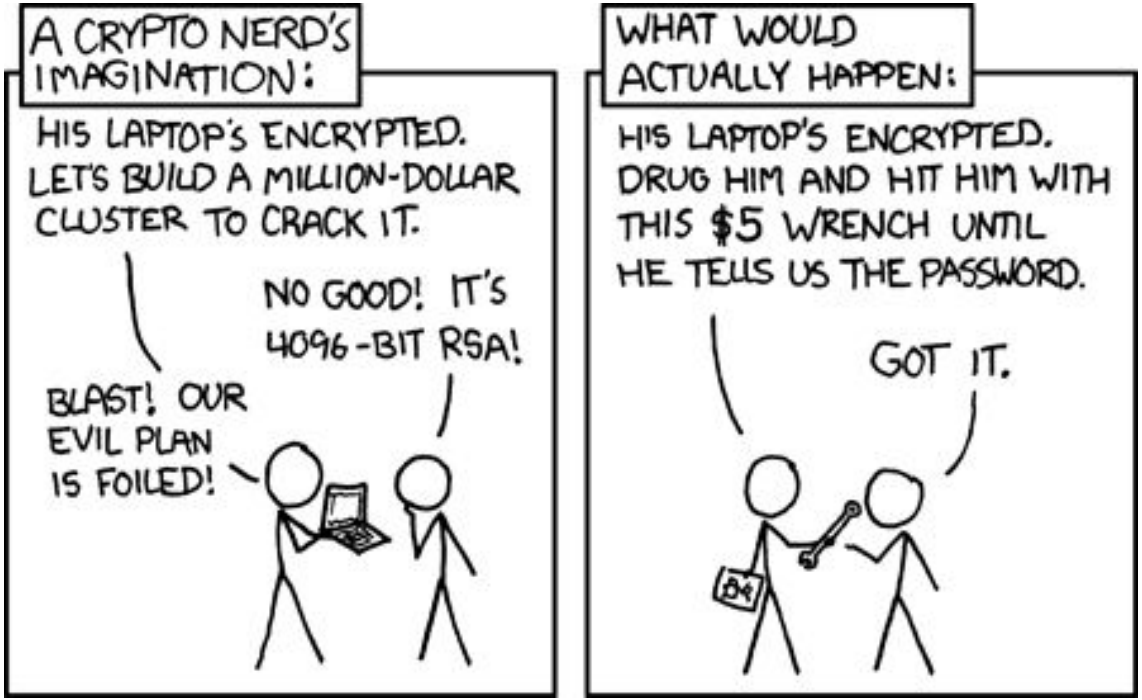
Vulnerability found by Tinkering

Vulnerability	Location	POC
VID-7: If you obtain a session ID, you can log in as another user without a password or OTP. Medium (CVSS 5.8) Spoofing	Lib\site-packages\django\contrib\sessions\backends\db.py > load() def load(self): s = self._get_session_from_db() return self.decode(s.session_data) if s else {}	1) Access https://ahnlab2.lge.com:8000/ with a web browser 2) Open developer tools -> Application -> Cookies 3) Enter session ID and value of other user session ID in cookies 4) Access https://ahnlab2.lge.com:8000/alpr with a web browser

Results and Conclusion

- It is difficult to tell security relevant failures from other functional failures. (Haenggi, Brian)
- It requires more experience to apply the attack techniques we have practiced in the course. (Seungkyu, Kibong)
- Choosing secure OSS packages reduce the risks from unintentional vulnerabilities. (Seongsik, Haenggi)
- Threat modeling is not only a set of technical activities, but also an opportunity to collaborate to build more secure systems. (Youngmi, Brian)
- You have to go through growing pains to learn security. (Jeff)

What would actually happen in our work



* Source: https://xkcd.com/538/

Q&A

What you would do differently if you start over?

- Next time we'd like to put more effort into exploitation than exploration.