

```
1 /* Snippet 1 */
2 char username[8];
3 int allow = 0;
4 printf("Enter your username: ");
5 gets(username);
6 if (grant_access(username)) {
7     allow = 1;
8 }
9 if (allow == 1) {
10     privileged_action();
11 }
```

- Purpose of the code:
- Vulnerability:
- Proposed fix:

```
1 /* Snippet 2 */
2 u_int nresp = packet_get_int();
3 if (nresp > 0) {
4     response = xmalloc(nresp * sizeof(char*));
5     for (i = 0; i < nresp; i++)
6         response[i] = packet_get_string(NULL);
7 }
```

- Purpose of the code:
- Vulnerability:
- Proposed fix:

```

1 /* Snippet 3 */
2 char *mail_auth(char *mechanism, authresponse_t resp, int argc, char *argv[])
3 {
4     char tmp[MAILTMPLLEN];
5     AUTHENTICATOR *auth;
6
7     /* make upper case copy of mechanism name */
8     ucase(strcpy(tmp, mechanism));
9     for (auth = mailauthenticators; auth; auth = auth->next)
10         if (auth->server && !strcmp(auth->name, tmp))
11             return (*auth->server) (resp, argc, argv);
12     return NIL; /* no authenticator found */
13 }

```

- Purpose of the code:
- Vulnerability:
- Proposed fix:

```

1 /* Snippet 4 */
2 char npath[MAXPATHLEN];
3 int i;
4
5 for (i = 0; *name != '\0' && i < sizeof(npath) - 1; i++, name++)
6 {
7     npath[i] = *name;
8     if (*name == '"')
9         npath[++i] == '"';
10 }
11 npath[i] = '\0'

```

- Purpose of the code:
- Vulnerability:
- Proposed fix:

```

1  /* Snippet 5 */
2  struct header {
3      unsigned int length;
4      unsigned int message_type;
5  };
6
7  char *read_packet(int sockfd) {
8      int n;
9      unsigned int length;
10     struct header hdr;
11     static char buffer[1024];
12
13     if (full_read(sockfd, (void *)&hdr, sizeof(hdr)) <= 0) {
14         error("full_read: %m");
15         return NULL;
16     }
17
18     length = ntohs(hdr.length);
19     if (length > (1024 + sizeof(struct header) - 1)) {
20         error("not enough room in buffer\n");
21         return NULL;
22     }
23
24     if (full_read(sockfd, buffer, length - sizeof(struct header)) <= 0) {
25         error("read: %m");
26         return NULL;
27     }
28
29     buffer[sizeof(buffer) - 1] = '\0';
30
31     return strdup(buffer);
32 }

```

- Purpose of the code:
- Vulnerability:
- Proposed fix: