Bibliography

Books

[1] Advanced Micro Devices. *AMD64 Architecture Programmer's Manual Volume 1: Application Programming*. Publication No. 24592, Revision 3.09. Advanced Micro Devices, Inc. Sunnyvale, CA, Sept. 2003 (cited on page 235).

[2] IBM Corporation. *FORTRAN Assembly Program for the IBM 7090*. 1961. URL: https://archive.computerhistory.org/resources/text/Fortran/102663110.05.01.acc.pdf (cited on page 257).

[3] ISO/IEC. *Information technology — Programming languages — C*. Standard. International Organization for Standardization, 2011 (cited on page 119).

[4] D. M. Ritchie K. Thompson. *UNIX Programmer's Manual (Fourth Edition)*. Available at https://dspinellis.github.io/unix-v3man/v3man.pdf. Nov. 1973 (cited on page 68).

[5] D. M. Ritchie K. Thompson. *UNIX Programmer's Manual (Third Edition)*. Available at https://dspinellis.github.io/unix-v3man/v3man.pdf. Feb. 1973 (cited on page 68).

[6] B. W. Kernighan. *UNIX: A History and a Memoir*. Independently published, Oct. 2019, page 197. ISBN: 978-1695978553 (cited on page 19).

[7] D. Knuth. *The Art of Computer Programming*. 3rd. Volume 1: Fundamental Algorithms. Addison–Wesley, 1997. Chapter 2.5, pages 435–456. ISBN: 0-201-89685-0 (cited on page 166).

[8] Steven Levy. *Hackers: Heroes of the Computer Revolution*. Garden City, NY: Anchor Press/-Doubleday, 1984. ISBN: 978-0-385-19195-1 (cited on page 12).

[9] C. E. Mackenzie. *Coded-Character Sets: History and Development*. Addison-Wesley Longman Publishing Co., Inc., 1980 (cited on page 31).

[10] M. K. McKusick and G. V. Neville-Neil. *The Design and Implementation of the FreeBSD Operating System*. 2nd. Addison-Wesley, 2014. ISBN: 978-0-321-96897-5 (cited on page 20).

[11] S. Meier. *Sid Meier's Memoir!: A Life in Computer Games*. W. W. Norton & Company, 2020. ISBN: 978-1324005872 (cited on page 187).

[12] E. I. Organick. *The Multics System: An Examination of Its Structure*. Cambridge, MA: MIT Press, 1972, pages xviii, 392. ISBN: 978-0-262-15012-5 (cited on page 19).

[13] D. Swade. *The History of Computing: A Very Short Introduction*. Oxford University Press, 2012 (cited on page 11).

[14] Tool Interface Standards (TIS) Committee. *Executable and Linking Format (ELF) Specification, Version 1.2*. UNIX International. 1995. URL: https://refspecs.linuxfoundation.org/elf/elf.pdf (cited on page 252).

[15] *UNIX System V Programmer's Manual*. First Edition. Holmdel, NJ, 1983 (cited on page 20).

[16] M. V. Wilkes. *Time-sharing Computer Systems (Third Edition)*. London: Macdonald and Jane's, 1974 (cited on page 62).

Articles

[17] Aleph One. "Smashing The Stack For Fun And Profit". In: *Phrack Magazine* 7.49 (Nov. 1996). URL: http://phrack.org/issues/49/14.html (cited on page 99).

[18] "Anecdotes". In: *IEEE Annals of the History of Computing* 3.3 (July 1981), pages 285–286 (cited on page 11).

[19] Crispin Bauer. "AppArmor: An Application Security System". In: *Proceedings of the 2007 Linux Security Summit*. LSS '07. Ottawa, Canada: USENIX Association, 2007, pages 2–2. URL: http://portal.acm.org/citation.cfm?id=1316548.1316550 (cited on page 84).

[20] Matt Bishop, Michael Dilger, et al. "Checking for race conditions in file accesses". In: *Computing systems* 2.2 (1996), pages 131–152 (cited on page 38).

[21] blexim. "Basic Integer Overflows". In: *Phrack Magazine* 11.60 (Dec. 2002). URL: https://phrack.org/issues/60/10#article (cited on page 188).

[22] S. R. Bourne. *Early days of Unix and design of sh*. https://www.youtube.com/watch?v=2kEJoWfobpA&t=2759s. June 2015 (cited on pages 57, 79).

[23] S. Bratus, M. E. Locasto, and M. L. Patterson. "Exploit programming: From buffer overflows to "weird machines" and theory of computation". In: (2011) (cited on page 128).

[24] S. Chazelas. *Absolute pathnames to commands in shell scripts (in reply to)*. https://groups.google.com/g/comp.unix.shell/c/z31RR92CnPk/m/nbl2pklbW-8J?hl=en. 2004 (cited on page 80).

[25] S. Chazelas. *CVE-2014-6271 (ShellShock)*. https://cve.mitre.org/cgi-bin/cvename.cgi?name=cve-2014-6271 (cited on page 80).

[26] D. Cohen. "On Holy Wars and a Plea for Peace". In: *IEEE Computer* 14.10 (1981), pages 48–54 (cited on page 237).

[27] F. J. Corbató. "On building systems that will fail". In: *ACM Turing award lectures*. Available at http://larch-www.lcs.mit.edu:8001/~corbato/turing91/. 2007, page 1990 (cited on page 62).

[28] F. J. Corbató, M. M. Daggett, and R. C. Daley. "An Experimental Time-Sharing System". In: *Proceedings of the Spring Joint Computer Conference*. May 1962, pages 335–344 (cited on page 19).

[29] F. J. Corbató, J. H. Saltzer, and V. A. Vyssotsky. *The Multics System: Multiplexed Information and Computing Service*. Technical Report FJCC Series. Cambridge, MA: MIT, GE, and Bell Labs, Oct. 1965, pages 185–196. URL: https://multicians.org/fjcc1.html (cited on page 19).

[30] J. Corbet. *Address space randomization in 2.6*. LWN.net. https://lwn.net/Articles/121845/. May 2005 (cited on page 161).

[31] C. Cowan et al. "StackGuard: Automatic Adaptive Detection and Prevention of Buffer-Overflow Attacks". In: *Proc. 7th USENIX Security Symposium*. San Antonio, TX, USA, Jan. 1998, pages 63–78. URL: https://www.usenix.org/legacy/publications/library/proceedings/sec98/cowan.html (cited on page 122).

[32] Solar Designer. *Getting around non-executable stack (and fix)*. https://seclists.org/bugtraq/1997/Aug/63. Aug. 1997 (cited on page 148).

[33] Solar Designer. *Linux kernel patch to remove stack exec permission*. Bugtraq mailing list. https://seclists.org/bugtraq/1997/Apr/31. Apr. 1997. (Visited on 08/18/2025) (cited on page 144).

[34] Solar Designer. "JPEG COM Marker Processing Vulnerability in Netscape browsers and Microsoft products, and a generic heap-based buffer overflow exploitation technique". In: (July 2000). https://www.openwall.com/articles/JPEG-COM-Marker-Vulnerability (cited on page 172).

[35] S. Dorward, R. Pike, and D. Ritchie. "The Inferno Operating System". In: *Bell Labs Technical Journal* 2.1 (1997), pages 5–18. URL: https://www.vitanuova.com/inferno/papers/bltj.pdf (cited on page 20).

[36] U. Drepper. *How To Write Shared Libraries*. https://akkadia.org/drepper/dsohowto.pdf. Aug. 2011 (cited on page 282).

[37] H. Etoh and K. Yoda. *ProPolice: Protecting from Stack-Smashing Attacks*. Technical report RT0371. IBM Research, Tokyo Research Laboratory, July 2000. URL: https://dominoweb.draco.res.ibm.com/reports/rt0371.pdf (cited on page 122).

[38] *Field splitting*. URL: https://pubs.opengroup.org/onlinepubs/000095399/utilities/xcu_chap02.html#:~:text=of%20informative%20text.-,2.6.5,-Field%20Splitting (cited on page 75).

[39] P. Frasunek. *WUFTPD 2.6.0 remote root exploit*. Bugtraq mailing list post. Accessed: 2025-08-17. June 2000. URL: https://marc.info/?l=bugtraq&m=96179429114160&w=2 (cited on page 127).

[40] A. Grabowski. *Add stack gap (random stack pointer)*. OpenBSD CVS commit message. https://cvsweb.openbsd.org/src/sys/kern/kern_exec.c.diff?r1=1.67&r2=1.68. Aug. 2001 (cited on page 161).

[41] F. T. Grampp and R. H. Morris. "UNIX operating system security". In: *AT&T Bell Laboratories Technical Journal* 63 (1984). Available at https://pages.cs.wisc.edu/~elisa/Grampp+Morris-UNIX-1984.pdf, pages 1649–1672 (cited on pages 59, 61–63, 68).

[42] S. B. Hamor. *[BUG] Vulnerability in PINE*. https://marc.info/?l=bugtraq&m=87602167419803&w=2. 1996 (cited on page 91).

[43] M. Hopkins. *A Summary of the 80486 Opcodes and Instructions*. USENET newsgroup post in `alt.lang.asm`. July 5, 1992. URL: https://gist.github.com/seanjensengrey/f971c20d05d4d0efc0781f2f3c0353da (visited on 08/02/2025) (cited on page 236).

[44] J. Hubicka. "ABI: register passing conventions change proposal". In: Nov. 2000. URL: https://web.archive.org/web/20140414124645/http://www.x86-64.org/pipermail/discuss/2000-November/001257.html (cited on page 245).

[45] *IEEE/Open Group Standard for Information Technology—Portable Operating System Interface (POSIX™) Base Specifications, Issue 8*. https://standards.ieee.org/ieee/1003.1/7700/. June 2024 (cited on page 20).

[46] Intel Corporation. *80386 High Performance 32-bit Microprocessor with Integrated Memory Management*. Technical report. Preliminary Datasheet, Order Number 231630-001. Santa Clara, CA: Intel Corporation, Oct. 1985 (cited on page 235).

[47] SRI International. *Improving the Security of your UNIX System*. Technical report. 1990. URL: https://nvlpubs.nist.gov/nistpubs/Legacy/IR/nistir4453.pdf (cited on page 62).

[48] W. Joy. *An Introduction to the C shell*. https://docs-archive.freebsd.org/44doc/usd/04.csh/paper.html (cited on page 50).

[49] *Kernel Support for miscellaneous Binary Formats (binfmt_misc)*. https://docs.kernel.org/admin-guide/binfmt-misc.html (cited on page 87).

[50] Thomas J. Killian. "Processes as Files". In: *Proceedings of the USENIX Summer Conference*. Salt Lake City, UT, USA: USENIX Association, 1984, pages 203–207 (cited on page 62).

[51] Anand Lal Shimpi. "Intel's 64-bit Xeon (Nocona) Launch". In: *AnandTech* (June 2004). URL: https://www.anandtech.com/show/1367 (visited on 08/04/2025) (cited on page 235).

[52] D. Lea. *A Memory Allocator*. Source distribution. 2000. URL: http://gee.cs.oswego.edu/pub/misc/malloc.c (cited on page 166).

[53] J. L. Lions. *Ariane 5 Flight 501 Failure: Report by the Inquiry Board*. Technical report. European Space Agency, July 1996. URL: https://www-users.cse.umn.edu/~arnold/disasters/ariane5rep.html (cited on page 187).

[54] H. Marco-Gisbert and I. Ripoll. "Exploiting Linux and PaX ASLR's Weaknesses on 32- and 64-bit Systems". In: *Black Hat Asia*. https://www.blackhat.com/docs/asia-16/materials/asia-16-Marco-Gisbert-Exploiting-Linux-And-PaX-ASLRs-Weaknesses-On-32-And-64-Bit-Systems.pdf. 2016 (cited on page 161).

[55] H. Marco-Gisbert and I. Ripoll. "Return-to-csu: A new method to bypass 64-bit Linux ASLR". In: *Black Hat Asia 2018*. 2018 (cited on page 155).

[56] S. Masheck. *The #! magic, details about the shebang/hash–bang mechanism on various Unix flavours*. https://www.in-ulm.de/~mascheck/various/shebang/. 2021 (cited on page 81).

[57] J. Mashey, D. Haight, and A. Glasser. *PWB Shell Manual*. https://www.in-ulm.de/~mascheck/bourne/PWB/. Bell Laboratories, May 1977 (cited on page 57).

[58] J. R. Mashey. “Using a Command Language as a High-Level Programming Language”. In: *Proceedings of the 2nd International Conference on Software Engineering*. Available at https://grosskurth.ca/bib/1976/mashey-command.pdf. San Francisco, California, USA, Oct. 1976, pages 169–176 (cited on page 68).

[59] M. Matz et al. *System V Application Binary Interface AMD64 Architecture Processor Supplement*. Specification. Draft Version 1.0. Available at https://gitlab.com/x86-psABIs/x86-64-ABI. The x86-64 ABI Group, July 2013. URL: https://gitlab.com/x86-psABIs/x86-64-ABI/-/raw/main/x86-64-psABI-1.0.pdf (visited on 08/02/2025) (cited on pages 235, 252).

[60] J. McDonald. *Defeating Solaris/SPARC non-executable stack protection*. https://seclists.org/bugtraq/1999/Mar/4. Mar. 1999 (cited on page 148).

[61] M. D. McIlroy. *A Research UNIX Reader: Annotated Excerpts from the Programmer’s Manual, 1971-1986*. Technical report CSTR 139. Available at https://www.cs.dartmouth.edu/~doug/reader.pdf. AT&T Bell Laboratories, 1987 (cited on page 29).

[62] J. Mckevitt and J. Bayliss. “A 16-bit HMOS microprocessor”. In: *1979 IEEE International Solid-State Circuits Conference. Digest of Technical Papers*. Volume XXII. IEEE, Feb. 1979, pages 168–169. DOI: 10.1109/ISSCC.1979.1155905 (cited on page 235).

[63] H. Meer, N. FitzGerald, and R. Temmingh. *Memory Corruption Attacks: The (Almost) Complete History*. Black Hat USA. https://media.blackhat.com/bh-us-10/whitepapers/Meer_FitzGerald_Temmingh/BlackHat-USA-2010-Meer-FitzGerald-Temmingh-Memory-Corruption-wp.pdf. 2010 (cited on page 99).

[64] Microsoft Corp. *Changes to Functionality in Microsoft Windows XP Service Pack 2, Part 3: Memory Protection Technologies*. https://learn.microsoft.com/previous-versions/windows/it-pro/windows-xp/bb457155(v=technet.10). Aug. 2004 (cited on page 141).

[65] Mindquest. “Grace Hopper and the First Computer Bug: How a Moth Changed the Future of Software Engineering”. In: *Mindquest* (2023). URL: https://mindquest.io/en/blog/news/8124/grace-hopper-and-the-first-computer-bug-how-a-moth-changed-the-future-of-software-engineering (visited on 08/23/2025) (cited on page 11).

[66] R. Morris and K. Thompson. “Password Security: A Case History”. In: *Communications of the ACM* 22.11 (Nov. 1979), pages 594–597 (cited on page 63).

[67] National Museum of American History. *Computer Bug Logbook*. https://americanhistory.si.edu/collections/object/nmah_334663. Accessed on August 23, 2025. Sept. 2025 (cited on page 11).

[68] nergal. “The advanced return-into-lib (c) exploits: Pax case study”. In: *Phrack Magazine* 11.54 (2001), pages 227–242 (cited on page 148).

[69] T. Newsham. *Format String Attacks*. Guardent R&D White Paper. Guardent, Inc., Sept. 2000. URL: https://seclists.org/bugtraq/2000/Sep/214 (cited on page 128).

[70] C. O’Donell and M. Sebor. *Field Experience With Annex K — Bounds Checking Interfaces*. Working paper N1967. ISO/IEC JTC1/SC22/WG14, Sept. 25, 2015. URL: https://www.open-std.org/jtc1/sc22/wg14/www/docs/n1967.htm (cited on page 119).

[71] Tavis Ormandy. *The GNU C library dynamic linker expands $ORIGIN in setuid library search path*. Full Disclosure Mailing List. [Online; accessed 20-Sep-2025]. Oct. 2010. URL: http://seclists.org/fulldisclosure/2010/Oct/257 (cited on page 84).

[72] N. Otterness. "Tiny ELF Files: Revisited in 2021". In: *nathanotterness.com* (Oct. 2021). URL: https://nathanotterness.com/2021/10/tiny_elf_modernized.html (cited on page 253).

[73] B. Page. *A report on the Internet Worm*. http://ftp.cerias.purdue.edu/pub/doc/morris_worm/worm.paper. Nov. 1988 (cited on page 116).

[74] PaX Team. *PaX: PAGEEXEC (old)*. https://pax.grsecurity.net/docs/pageexec.old.txt. (accessed 2025-08-18). grsecurity, Nov. 2000 (cited on page 144).

[75] PaX Team. *PaX Address-Space Layout Randomization*. https://pax.grsecurity.net/docs/aslr.txt. Accessed 2025-08-17. July 2001 (cited on page 161).

[76] PaX Team. *PaX: PAGEEXEC*. https://pax.grsecurity.net/docs/pageexec.txt. (accessed 2025-08-18). grsecurity, Mar. 2003 (cited on page 144).

[77] PaX Team. *PaX: SEGMEXEC*. https://pax.grsecurity.net/docs/segmexec.txt. (accessed 2025-08-18). grsecurity, May 2003 (cited on page 145).

[78] Phantasmal Phantasmagoria. *The Malloc Maleficarum*. Bugtraq mailing list. Oct. 2005. URL: https://seclists.org/bugtraq/2005/Oct/118 (cited on page 174).

[79] R. Pike and K. Thompson. *Plan 9 from Bell Labs*. Technical Report. Bell Labs, 1995. URL: http://plan9.bell-labs.com/sys/doc/9.html (cited on page 20).

[80] M. W. Pirtle, W. Lichtenberger, and B. Lampson. *Project Genie: Berkeley Timesharing System*. Technical Report. University of California, Berkeley, 1964. URL: https://en.wikipedia.org/wiki/Project_Genie (cited on page 19).

[81] T. de Raadt. *OpenBSD 3.3 Release*. OpenBSD Announcement. https://www.openbsd.org/33.html. May 2003 (cited on page 141).

[82] B. Raiter. *A Whirlwind Tutorial on Creating Really Teensy ELF Executables for Linux*. https://www.muppetlabs.com/~breadbox/software/tiny/teensy.html. 1999 (cited on page 253).

[83] J. Reynolds. *RFC 1135: The Helminthiasis of the Internet*. https://www.rfc-editor.org/rfc/rfc1135. Dec. 1989 (cited on page 116).

[84] D. Ritchie. *Dennis Ritchie and Hash-Bang*. https://www.talisman.org/~erlkonig/documents/dennis-ritchie-and-hash-bang.shtml (cited on page 80).

[85] D. M Ritchie. "UNIX Time-Sharing System: A Retrospective". In: *Bell System Technical Journal* 57.6 (1978), pages 1947–1969 (cited on page 19).

[86] D. M. Ritchie. "On the security of UNIX". In: *The UNIX Programmer's Manual (Sixth Edition)* (1975). Available at https://github.com/manjunath5496/Dennis-M-Ritchie-papers/blob/master/ist(8).pdf (cited on page 59).

[87] D. M. Ritchie. "On the security of UNIX". In: *The UNIX Programmer's Manual (Seventh Edition)* (1979). Available at https://maibriz.de/unix/ultrix/etc/security.pdf (cited on page 59).

[88] D. M. Ritchie. "The UNIX system: The evolution of the UNIX time-sharing system". In: *AT&T Bell Laboratories Technical Journal* 63.8 (1984). Available at https://www.nokia.com/bell-labs/about/dennis-m-ritchie/hist.pdf, pages 1577–1593 (cited on page 46).

[89] D. M. Ritchie. *The UNIX Time-sharing system (DRAFT)*. https//www.tuhs.org/Archive/Distributions/Research/McIlroy_v0/UnixEditionZero-Threshold_OCR.pdf (cited on page 34).

[90] D. M. Ritchie and K. Thompson. "The UNIX time-sharing system". In: *Bell System Technical Journal* 57.6 (1978), pages 1905–1929 (cited on page 19).

[91] J. A. Rochlis and M. W. Eichin. "With microscope and tweezers: the worm from MIT's perspective". In: *Commun. ACM* 32.6 (June 1989), pages 689–698. ISSN: 0001-0782. DOI: 10.1145/63526.63528. URL: https://doi.org/10.1145/63526.63528 (cited on page 116).

[92] R. Roemer et al. "Return-Oriented Programming: Systems, Languages, and Applications". In: 15.1 (Mar. 2012). ISSN: 1094-9224. DOI: 10.1145/2133375.2133377 (cited on page 148).

[93] rusty and soren. *Symlink problem (Tested only on a Digital Unix 4.0)*. https://cliplab.org/~alopez/bugs/bugtraq9/0019.html. Accessed: 2024-08-27. Apr. 1997 (cited on page 38).

[94] A. Scherr. *Oral History Interview*. Computer History Museum. Admitted printing CTSS password file to bypass time limits in 1962. Nov. 2012 (cited on page 62).

[95] M. S. Schlansker and B. R. Rau. "EPIC: Explicitly Parallel Instruction Computing". In: *IEEE Computer* 33.2 (Feb. 2000), pages 37–45. DOI: 10.1109/2.820037 (cited on page 235).

[96] R. Seacord, D. Svoboda, and W. Dormann. "Memory Errors: The Past, the Present, and the Future". In: *Technical Report CMU/SEI-2012-TN-013* (2012). https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=8419 (cited on page 99).

[97] *sh – shell, the standard command language interpreter*. https://pubs.opengroup.org/onlinepubs/009695399/utilities/sh.html. 2001 (cited on page 87).

[98] H. Shacham. "The geometry of innocent flesh on the bone: return-into-libc without function calls (on the x86)". In: *Proceedings of the 14th ACM Conference on Computer and Communications Security*. CCS '07. Alexandria, Virginia, USA: Association for Computing Machinery, 2007, pages 552–561. ISBN: 9781595937032. DOI: 10.1145/1315245.1315313 (cited on page 148).

[99] H. Shacham et al. "On the Effectiveness of Address-Space Randomization". In: *Proc. 11th ACM Conference on Computer and Communications Security (CCS)*. 2004, pages 298–307. DOI: 10.1145/1030083.1030124 (cited on page 162).

[100] Shooting Shark. "Nasty Unix Tricks". In: *Phrack Magazine* 1.6 (Apr. 1986). URL: https://phrack.org/issues/6/5#article (cited on page 61).

[101] Software Engineering Institute. *SEI CERT C Coding Standard*. 2016. URL: https://wiki.sei.cmu.edu/confluence/display/c/SEI+CERT+C+Coding+Standard (visited on 02/27/2024) (cited on page 118).

[102] I. L. Taylor. *Linker relro*. https://www.airs.com/blog/archives/189. Aug. 2008 (cited on page 177).

[103] The Jargon File. *Bug*. http://www.catb.org/jargon/html/B/bug.html. Accessed on August 23, 2025. Aug. 2025 (cited on page 11).

[104] Ken Thompson. "The Unix Command Language". In: *Infotech State of the Art Report: Structured Programming*. Maidenhead, England: Infotech International, 1976 (cited on pages 60, 68).

[105] L. B. Torvalds. *What would you like to see most in minix?* https://groups.google.com/g/comp.os.minix/c/dlNtH7RRrGA/m/SwRavCzVE7gJ. Aug. 1991 (cited on page 20).

[106] T. Twillman. *Exploit for proftpd 1.2.0pre6*. Bugtraq mailing list post. Accessed: 2025-08-17. Sept. 1999. URL: https://seclists.org/bugtraq/1999/Sep/328 (cited on page 128).

[107] T. Van Vleck. *How the Air Force cracked Multics Security*. https://www.multicians.org/security.html. 1994 (cited on page 63).

[108] A. van de Ven. *Linux: add brk heap randomization*. Linux kernel commit 2a77f7b. https://git.kernel.org/linus/2a77f7b. Apr. 2008 (cited on page 161).

[109] *Windows Vista ASLR Announcement*. Microsoft Security Development Lifecycle presentation, ph-neutral. 2006 (cited on page 161).

[110] R. Wojtczuk. *Defeating Solar Designer's Non-executable Stack Patch*. Bugtraq mailing list. https://insecure.org/sploits/non-executable.stack.problems.html. Jan. 1998 (cited on page 144).

[111] M. Zalewski. *Delivering Signals for Fun and Profit: Understanding, exploiting and preventing signal-handling related vulnerabilities*. Technical report. Accessed: 2025-08-28. BindView Corporation, 2001. URL: https://lcamtuf.coredump.cx/signals.txt (cited on page 38).

URLs

[112] Gallopsled et al. *pwntools*. URL: https://docs.pwntools.com/en/stable/ (cited on pages 102, 109, 236).

[113] davi942j. *one_gadget*. URL: https://github.com/david942j/one%5C_gadget (cited on page 159).

[114] FreeBSD Project. *FreeBSD Release and Documentation*. URL: https://www.freebsd.org (cited on page 20).

[115] A. Griffiths. *Phoenix Virtual Machine*. URL: http://exploit.education/phoenix/ (cited on page 99).

[116] W. Joy. *Berkeley Software Distribution (BSD) Unix*. 1977. URL: https://cgit.freebsd.org/src/tree/share/misc/bsd-family-tree (cited on page 20).

[117] NetBSD Project. *NetBSD Release and Portability Goals*. URL: https://www.netbsd.org (cited on page 20).

[118] OpenBSD Project. *OpenBSD Release and Documentation*. URL: https://www.openbsd.org (cited on page 20).

[119] Z. Riggle. *pwndbg*. 2025. URL: https://pwndbg.re/ (cited on page 13).

[120] S. Schirra. *Ropper*. URL: https://github.com/sashs/Ropper (cited on page 152).

[121] *The GNU Project*. Free Software Foundation. 2025. URL: https://www.gnu.org/ (cited on page 20).

[122] L. B. Torvalds and The Linux Development Community. *The Linux Kernel*. 2025. URL: https://www.kernel.org (cited on page 20).

[123] *UTM*. 2025. URL: https://mac.getutm.app/ (cited on page 12).