Appendix A – an exemplary snippet of the original data

	column_name	type	head_values
1	SYSCALL_timestamp	int	0 0 0 0 0 0 0 0 0 0 ...
2	SYSCALL_arch	chr	"aarch64" "aarch64" "aarch64" ...
3	SYSCALL_syscall	chr	"mmap" "clone" "munmap" ...
4	SYSCALL_success	chr	"yes" "yes" "yes" "yes" ...
5	SYSCALL_exit	num	5.48e+11 6.18e+05 0.00 0.00 0.00 ...
6	PROCESS_comm	chr	"apache2" "apache2" "apache2" ...
7	PROCESS_exe	chr	"/usr/sbin/apache2" "/usr/sbin/apache2" ...
8	PROCESS_PATH	chr	"systemd>apache2" "systemd>apache2" ...
9	CUSTOM_openFiles	chr	"[]" "[]" "[]" ...
10	CUSTOM_libs	chr	"[]" "[]" "[]" ...
11	PROCESS_uid	chr	"www-data" "www-data" "www-data" ...
12	PROCESS_gid	chr	"www-data" "www-data" "www-data" ...
13	SYSCALL_exit_hint	chr	"548108500992" "618484" "0" ...
14	SYSCALL_pid	int	584020 584020 584020 618484 618484 ...
15	USER_AUTH	int	0 0 0 0 0 0 0 0 0 0 ...
16	USER_MGMT_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
17	CRED_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
18	USER_ERR_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
19	USYS_CONFIG_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
20	CHID_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
21	SELINUX_ERR_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
22	SYSTEM_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
23	SERVICE_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
24	DAEMON_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
25	NETFILTER_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
26	SECCOMP_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
27	AVC_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
28	ANOM_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
29	INTEGRITY_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
30	KERNEL_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
31	RESP_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
32	SELINUX_MGMT_COUNT	int	0 0 0 0 0 0 0 0 0 0 ...
33	CUSTOM_openSockets	chr	"[]" "[]" "[]" ...
34	USER_ACTION_op	chr	"" "" "" "" ...
35	USER_ACTION_src	chr	"" "" "" "" ...
36	USER_ACTION_res	chr	"" "" "" "" ...
37	USER_ACTION_addr	chr	"" "" "" "" ...
38	PROCESS_name	chr	"/usr/sbin/apache2" "/usr/sbin/apache2" ...
39	KILL_process	logi	NA NA NA NA NA NA ...
40	KILL_uid	logi	NA NA NA NA NA NA ...

Appendix B – process graph visualization constructed for the original data

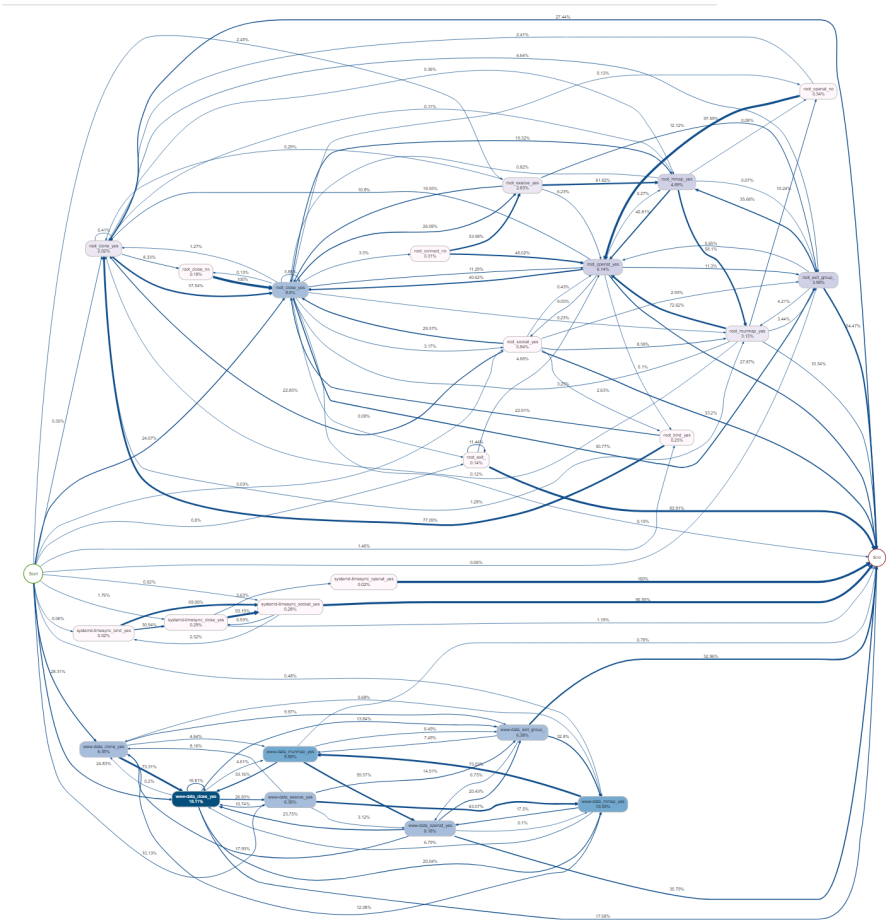

Fig. 4. A visualization of a process graph based on process traces that jointly cover only 80% of our process discovery part of data. The graph was constructed using a method implemented in the bupaR library for R language [17]. In practice, such a complex graph is difficult to interpret and utilize for the extraction of useful knowledge.